MLB-R130G-01

HITACHI

日立産業用コンピュータ

HF-BX･BT/LX シリーズ

HF-BX1000/LX
HF-BX1200/LX
HF-BT1000/LX

RAS 機能マニュアル

ユーザーズ
マニュアル

MLB-R130G-01

HITACHI

日立産業用コンピュータ

HF-BX・BT/LX シリーズ

HF-BX1000/LX
HF-BX1200/LX
HF-BT1000/LX

RAS 機能マニュアル

マニュアルはよく読み、保管してください。

- 操作を行う前に、安全上の指示をよく読み、十分理解してください。
- このマニュアルは、いつでも参照できるよう、手近なところに保管してください。

ユーザーズ
マニュアル

この製品を輸出される場合には、外国為替及び外国貿易法の規制
並びに米国輸出管理規則など外国の輸出関連法規をご確認のうえ、
必要な手続きをお取りください。
なお、不明な場合は、弊社担当営業にお問い合わせください。

２０１３年　１０月　（第１版）　ＭＬＢ－Ｒ１３０Ｇ－００
２０１３年　１１月　（第２版）　ＭＬＢ－Ｒ１３０Ｇ－０１

● このマニュアルの一部または全部を無断で転写したり複写したりすることは、
固くお断りいたします。
● このマニュアルの内容を、改良のため予告なしに変更することがあります。

All Rights Reserved, Copyright © 2013 Hitachi Information & Control Solutions, Ltd.

このページは白紙です。

⚠ 安全にお取り扱いいただくために

装置を操作する前に、以下に述べられている安全上の説明をよく読み、十分理解してください。

- 操作は、このマニュアル内の指示、手順に従って操作してください。
- 装置やマニュアルに表示されている安全に関する注意事項は特に注意を払い、必ず守ってください。これを怠ると、人身上の傷害や装置を含む財産の破損を引き起こす恐れがあります。
- 安全に関する注意事項は、下に示す見出しによって表示されます。これは安全警告記号と「危険」、「警告」、「注意」、および「通知」という見出し語を組み合わせたものです。

> ⚠ これは安全警告記号です。人への危害を引き起こす潜在的な危険に注意を喚起するために用いられます。起こりうる傷害または死を回避するためにこのシンボルの後に続く安全に関するメッセージに従ってください。
>
> ⚠ **危　険**：死亡または重大な傷害を引き起こす可能性が高い差し迫った危険の存在を示すのに用いられます。
>
> ⚠ **警　告**：死亡または重大な傷害を引き起こすかもしれない潜在的な危険の存在を示すのに用いられます。
>
> ⚠ **注　意**：軽度の傷害または中程度の傷害を引き起こす恐れのある潜在的な危険の存在を示すのに用いられます。

> **通　知**：これは、人身傷害とは関係のない損害をひき起すおそれのある危険の存在を示すのに用いられます。

なお、「留意事項」という見出し語は、装置の取扱いおよび操作上の注意書きを示すのに用いられます。

- マニュアルに記載されている以外の操作は行わないでください。装置について何か問題がある場合は、保守員をお呼びください。
- 装置を操作する前に、このマニュアルをよく読み、書かれている指示や注意を十分に理解してください。
- このマニュアルは、必要なときにすぐ参照できるよう、使いやすい場所に保管してください。
- 装置やマニュアルに表示されている注意事項は、十分に検討されたものでありますが、それでも、予測を超えた事態が起こることが考えられます。操作に当たっては、指示に従うだけでなく、常に自分自身でも注意するようにしてください。

⚠ 安全にお取り扱いいただくために（続き）

1．共通的な注意事項

以下に述べられている安全上の説明をよく読み、十分理解してください。

1．1 ⚠ 警告

- この装置は、人命に直接関わる安全性を要求されるシステムに適用する目的で製造されたものではありません。このような用途に使用する可能性がある場合には、弊社営業窓口へ相談してください。

- 万一、発煙・異臭などがあった場合は、装置への給電を遮断し、電源ケーブルをコンセントから抜いて、お買い求め先または保守員に連絡してください。故障状態のままお使いになると火災や感電の原因になります。

- この装置は磁気ディスク装置を内蔵しています。故障の原因になりますので、打撃、衝撃、振動を与えないよう取り扱ってください。万一、この装置を落としたりケースを破損した場合は、電源ケーブルをコンセントから抜いてから、保守員に連絡してください。そのまま使用すると、火災や感電の原因になります。また、開梱時および持ち運ぶ時は衝撃を与えないでください。

- 火災や感電の原因になりますので、この装置を改造しないでください。また、改造により発生した結果については、一切責任を負いかねますので、あらかじめご了承ください。

１．２ ⚠ 注意

- 装置が落下または転倒するとけがをする恐れがあります。装置の運搬時には十分注意してください。

- この装置を開梱する時や持ち運ぶ時に指をはさんだり、ぶつけたりしてけがをしないように注意してください

１．３　通知

- この装置単独ではシステムの安全は確保できません。この装置が万一故障したり誤動作やプログラムに欠陥があった場合でも、ご使用されるシステムの安全が十分に確保されるよう、人身事故・重大な災害に対する安全対策のための保護・安全回路を外部に設けるなど、システム的な保護処置を施した設計としてください。

- 弊社指定の基本ソフトウェアを使用してください。それ以外の基本ソフトウェアを使用した場合の動作は、保証できません。

- 故障の原因になりますので、この装置をじんあいの多い所や、腐食性ガスのある環境で使用することは避けてください。

- この装置の底面および背面には、吸排気孔のためのスペースを必ず確保してください。温度上昇による故障または短寿命の原因になります。また、保守作業のための保守スペースが必要です。

- 非常遮断（OSのシャットダウン処理を行わずに電源ケーブルをコンセントから抜いたり、ブレーカを切ったりすること）をすると、OSやアプリケーションが正常に動作しなくなったり、保存データの安全性に問題が生じたりすることがあります。何らかの異常が発生し緊急に遮断しなければならないとき以外は、絶対に実施しないでください。

- 電源の供給元で電源を遮断した場合、システムを自動で復旧できないことがありますので、注意してください。

はじめに

このマニュアルは、日立産業用コンピュータHF-BX･BT/LXシリーズ（以下、この装置と称す）のRAS（Reliability, Availability, Serviceability）機能の使用方法について記述したものです。

＜マニュアルの構成＞

このマニュアルは、次のような構成となっています。

第１章　RAS機能でできること
第２章　RAS機能の監視対象
第３章　RAS機能の設定
第４章　ハードウェア状態の確認
第５章　ハードウェアの制御
第６章　ライブラリ関数
第７章　保守・障害解析関連
第８章　インストール方法

＜RAS機能をご使用になる際の注意事項＞

- RAS機能のサポート範囲について
  この装置のRAS機能のサポート範囲については、「第１章　RAS機能でできること」をご覧ください。

- RAS外部接点インターフェースについて
  このマニュアルの説明の中で、RAS外部接点インターフェースに関する説明がありますが、このインターフェースはHF-BX/LXのみとなります。

＜記憶容量の計算値についての注意＞

- 2^n計算値の場合（メモリ容量・所要量、ファイル容量・所要量など）

  1KB（キロバイト）＝1,024バイトの計算値です。

  1MB（メガバイト）＝1,048,576バイトの計算値です。

  1GB（ギガバイト）＝1,073,741,824バイトの計算値です。

- 10^n計算値の場合（ディスク容量など）

  1KB（キロバイト）＝1,000バイトの計算値です。

  1MB（メガバイト）＝$1,000^2$バイトの計算値です。

  1GB（ギガバイト）＝$1,000^3$バイトの計算値です。

＜商標について＞

・LINUXは米国及びその他の国におけるLinus Torvaldsの登録商標です。

・RED HATと Shadowman logoは米国およびそのほかの国において登録されたRed Hat, Inc.の商標です。

・インテル、Intel、Intel Core、Celeron は米国およびその他の国におけるIntel Corporationの商標です。

・Wake on LANは米国IBM Corporationの商標です。

・Magic PacketはAdvanced Micro Devices, Inc.の商標または登録商標です。

・上記以外にこのマニュアルに記載されている他社製品名（ソフトウェア、ハードウェア）は、各社の登録商標、商標、または商品です。

目次

⚠安全にお取り扱いいただくために S-1

第１章　RAS機能でできること 1-1

１．１　HF-BX･BT/LXシリーズ RASユーティリティパッケージ 1-1

１．２　概要 1-2

第２章　RAS機能の監視対象 2-1

２．１　ファン監視機能 2-1

２．２　筐体内温度監視機能 2-2

２．３　ドライブ障害予測機能（SMART監視） 2-3

２．４　ドライブ使用時間監視機能 2-4

２．５　OSロック監視機能 2-5

２．６　ウォッチドッグタイマ監視機能 2-6

２．６．１　ユーザプログラムの動作状態監視に使用する方法 2-6

２．７　RAID監視機能　（Dモデルのみ） 2-8

２．７．１　RAIDの状態遷移について 2-9

２．７．２　メディアエラーについて 2-10

第３章　RAS機能の設定 3-1

３．１　RAS機能設定コマンド（rasconfig） 3-1

３．１．１　概要 3-1

第４章　ハードウェア状態の確認 4-1

４．１　ハードウェア状態表示機能 4-2

４．１．１　RAS状態表示コマンド（rasstat） 4-2

４．２　RASイベント情報管理機能 4-6

４．２．１　概要 4-6

４．２．２　RASイベントハンドリング概要 4-6

４．２．３　RASイベントログ情報保存 4-6

４．３　状態表示デジタルLED機能 4-7

４．３．１　概要 4-7

４．３．２　表示するステータスコード 4-8

４．３．３　ステータスコードの表示 4-12

４．３．４　状態表示デジタルLED制御関数 4-13

４．４　内蔵ドライブ監視機能 4-14

４．４．１　ドライブ障害予測監視機能 4-14

４．４．２　ドライブ使用時間監視機能 4-15
４．５　RASリモート通知機能（SNMP） 4-17
４．５．１　概要 4-17
４．５．２　SNMP拡張MIB一覧 4-18
４．５．３　hfwLxサブツリー 4-19
４．５．４　hfwExMibInfoサブツリー 4-19
４．５．５　hfwLxExMibサブツリー 4-19
４．５．６　hfwRasStatusサブツリー 4-19
４．５．７　hfwRasSettingサブツリー 4-24
４．５．８　hfwRasInfoサブツリー 4-24
４．５．９　hfwLxMibTrapサブツリー 4-25
４．５．10　hfwRasErrorTrapサブツリー 4-25
４．５．11　hfwRasRecoverTrapサブツリー 4-26
４．５．12　hfwRasInfoTrapサブツリー 4-26
４．５．13　RAS状態照会機能 4-27
４．５．14　イベント通知機能 4-28
４．５．15　RASリモート通知機能の開始手順 4-29
４．６　RASライブラリ（GetRasStat関数、WaitForRasEvent関数）による状態取得 4-31

第５章　ハードウェアの制御 5-1
５．１　装置の自動シャットダウン 5-2
５．１．１　ファン異常検出による自動シャットダウン 5-2
５．１．２　高温異常検出による自動シャットダウン 5-3
５．１．３　リモートシャットダウン入力検出による自動シャットダウン（HF-BX/LXのみ） 5-3
５．２　RASライブラリによる制御 5-4

第６章　ライブラリ関数 6-1
６．１　RASライブラリインタフェース 6-1
６．１．１　概要 6-1
６．１．２　RAS汎用接点の出力制御関数（GendoControl）（HF-BX/LXのみ） 6-2
６．１．３　RAS汎用接点の信号入力状態取得関数（GetGendi）（HF-BX/LXのみ） 6-4
６．１．４　状態表示デジタルLED制御関数（SetStCode7seg、TurnOff7seg、SetMode7seg） 6-5
６．１．５　RAS状態取得関数（GetRasStat） 6-8
６．１．６　ウォッチドッグタイマ制御関数（WdtControl） 6-13
６．１．７　RASイベント情報取得関数（WaitForRasEvent） 6-14
６．１．８　サンプルプログラム 6-18

第７章　保守・障害解析関連 7-1
７．１　メモリダンプ機能 7-1
７．１．１　概要 7-1
７．１．２　メモリダンプの採取要因 7-2
７．１．３　メモリダンプファイル保存仕様 7-2
７．２　ログファイル収集機能 7-3
７．２．１　概要 7-3
７．２．２　raslogsaveコマンド概要 7-3
７．２．３　raslogsaveコマンド使用方法 7-4
７．２．４　ログファイル収集コマンド（raslogsave） 7-5
７．３　保守操作支援コマンド 7-8
７．３．１　RASイベントログ表示コマンド（raslogpt） 7-9
７．３．２　メディアエラー時通知設定機能 (rasraidctl NOTIFY)(Dモデルのみ) 7-10
７．４　RAS障害シミュレーション機能 7-12
７．４．１　概要 7-12
７．４．２　RAS障害シミュレーションコマンド（rassimtool） 7-13

第８章　インストール方法 8-1
８．１　概要 8-1
８．２　RASユーティリティパッケージのインストール 8-1
８．３　RASソフトウェア機能で使用する設定ファイルとOSカスタマイズ用設定ファイル 8-2

図目次

図２－１　ユーザプログラムの動作状態監視処理の例 2-7
図２－２　RAID状態遷移（Dモデルのみ） 2-10
図４－１　状態表示デジタルLED 4-7
図４－２　ハードウェアステータスコード 4-8
図４－３　アプリケーションステータスコード 4-10
図４－４　STOPエラーコード 4-10
図４－５　ウォッチドッグタイマタイムアウトコード 4-11
図４－６　サブツリー 4-18
図４－７　状態照会時のネットワーク構成例 4-27
図４－８　イベント通知機能ネットワーク構成例 4-28
図６－１　GENDO接点の動作（HF-BX/LXのみ） 6-3

表目次

表１－１　RASユーティリティパッケージ型式一覧 1-1
表２－１　ファン監視における装置の状態とMCALL接点の動作 2-1
表２－２　筐体内温度監視における装置の状態とMCALL接点の動作 2-2
表２－３　RAID状態とMCALL接点の動作 2-9
表２－４　RAID状態とその意味 2-10
表３－１　rasconfigコマンドのエラーメッセージ一覧 3-1
表４－１　rasstatコマンドで表示される状態表示項目（詳細表示） 4-4
表４－２　rasstatコマンドで表示される状態表示項目（簡易表示） 4-5
表４－３　RASイベントログファイル仕様 4-6
表４－４　ハードウェアステータスコード一覧 4-9
表４－５　ステータスコード表示条件一覧 4-12
表４－６　ステータスコードの表示種別一覧 4-12
表４－７　状態表示デジタルLED制御関数一覧 4-13
表４－８　内蔵ドライブ監視機能概略仕様 4-14
表４－９　障害予測監視対象ドライブ一覧 4-14
表４－10　HDD障害予測監視機能の動作仕様 4-14
表４－11　ドライブ状態毎の障害予測監視 4-15
表４－12　ドライブ使用時間監視対象ドライブ一覧 4-15
表４－13　ドライブ使用時間監視機能の動作仕様 4-15
表４－14　ドライブ状態毎の使用時間更新条件 4-16
表４－15　RASリモート通知機能（SNMP）概略仕様 4-17
表４－16　hfwExMibInfo下サブツリー 4-19
表４－17　hfwRasStatus下サブツリー 4-19
表４－18　hfwRasSetting下サブツリー 4-24
表４－19　hfwRasInfoサブツリー 4-24
表４－20　hfwRasErrorTrap下サブツリー 4-25
表４－21　hfwRasRecoverTrap下サブツリー 4-26
表４－22　hfwRasInfoTrap下サブツリー 4-26
表４－23　照会情報一覧 4-27
表４－24　RAS状態照会機能の動作仕様 4-27
表４－25　トラップ通知一覧 4-28
表６－１　RASライブラリ提供関数一覧 6-1
表６－２　GetRasStat構造体メンバ一覧 6-10
表６－３　WaitForRasEvent構造体メンバ一覧 6-15
表６－４　パラメータ格納値 6-15
表６－５　サンプルプログラム一覧 6-18
表７－１　メモリダンプを採取する要因一覧 7-2

表７－２　メモリダンプファイル保存仕様 7-2
表７－３　保守操作支援コマンド一覧 7-8
表７－４　rasraidctl　NOTIFYコマンドのエラーメッセージ一覧 7-11
表７－５　rassimtoolコマンドの[模擬部位]、[模擬内容]オプション一覧 7-14
表７－６　RAIDアレイの状態 組合せ表 7-15
表７－７　rassimtoolコマンドのエラーメッセージ一覧 7-16
表８－１　RASソフトウェア機能の設定ファイル一覧 8-2

第１章　RAS機能でできること

HF-W/LXおよびHF-BX･BT/LXシリーズは、高信頼な産業用コンピュータとしてのRAS（Reliability, Availability, Serviceability）機能を備えています。

この章では、RASユーティリティパッケージをインストールすることで組み込まれるRAS機能について説明します。

１．１　HF-BX･BT/LXシリーズ RASユーティリティパッケージ

RAS機能および障害解析機能は、以下のソフトウェアにてサポートします。

表１－１　RASユーティリティパッケージ型式一覧

No.	パッケージ名称	型式	サポートOS	備考
1	RASユーティリティパッケージ	CS-LXR81G	Red Hat Enterprise Linux Workstation 6.4（x86 32bit 日本語）	HF-BX1000/LX HF-BX1200/LX HF-BT1000/LX

留意事項

・本装置は工場出荷時にRASユーティリティパッケージをインストールしております。ご使用の際に改めてインストールする必要はありません。

・RASユーティリティパッケージは、上記型式とサポートOSおよびHF-BX･BT/LXシリーズとの組み合わせでのみ使用できます。

異なるOSおよびHF-BX･BT/LXシリーズ以外との組み合わせはサポート対象外となります。

１．２　概要

以下にRAS機能の概要について説明します。

（１）メモリダンプ採取機能

この装置が予期せずに停止してしまった場合など障害が発生したときに、キーボード操作（Alt + SysRq + c キー）によって、システムメモリの内容をファイル（メモリダンプファイル）に記録します。このメモリダンプの内容を解析することによって、障害の原因を調査することができます。（「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「７．２ メモリダンプ採取機能」参照）

（２）自動シャットダウン機能

ファン異常や筐体内温度異常、リモートシャットダウン信号入力を検出した場合に自動的にシャットダウンを実施します（５．１　装置の自動シャットダウン 参照）。ファン異常や筐体内温度異常時に自動的にシャットダウンを実施するかどうかは、「rasconfig コマンド」を使用して設定できます。（３．１　RAS機能設定コマンド（rasconfig）参照）

出荷時の自動シャットダウン設定は以下のとおりです。

・ファン異常発生時：自動的にシャットダウンします。
・筐体内温度異常発生時：自動的にはシャットダウンしません。
・リモートシャットダウン信号入力時：自動的にシャットダウンします。（HF-BX/LXのみ）

（３）ウォッチドッグタイマ監視機能

この装置に実装されているウォッチドッグタイマを使用するためのライブラリを提供します。
また、高レベルのプロセス優先度において自動的にウォッチドッグタイマのリトリガを行い、プロセスが正常にスケジューリングされていることを監視する機能も提供します。この機能を使用するかどうかは、「rasconfig コマンド」を使用して設定できます。（３．１　RAS機能設定コマンド（rasconfig）参照）

（４）RAS状態表示

この装置のファンや筐体内温度の状態を「rasstat コマンド」を使用して確認できます。
（４．１．１　RAS状態表示コマンド（rasstat）参照）

（５）RAS外部接点制御機能

オプションのRAS外部インタフェースには、ユーザが使用できる外部接点が入力用に2点と出力用に2点ずつ用意されています。これらの接点を使用して外部機器からの信号をこの装置に入力したり、この装置から外部に信号を出力したりすることができます。ユーザがこれらの接点を使用するためのライブラリ関数を提供します。（第６章　ライブラリ関数 参照）

（６）ドライブ障害予測機能（SMART）

SMARTは、Self-Monitoring, Analysis and Reporting Technologyの略でHDD自身がドライブ内の各種情報を監視して信頼性を評価する機能です。（２．３　ドライブ障害予測機能（SMART監視） 参照）この機能を使用することで、ドライブの障害を予測することができます。ただし、以下の点に注意してください。

・SMARTは、すべての障害を予測することはできません。そのため、SMARTによる障害予測がされる前にドライブが故障する場合もあります。
・SMARTの障害予測が発生した際は、速やかにデータのバックアップおよびドライブの交換を行ってください。
・SMARTの障害予測が発生した際、その原因を解析することはできません。

（７）RAS障害シミュレーション機能

この装置のハードウェア状態をシミュレートします。これによって、実際にハードウェア異常が発生していなくてもユーザアプリケーションのテストを実施することができます。
（７．４　RAS障害シミュレーション機能 参照）

（８）RASイベントハンドリング機能

ファン異常、筐体内温度異常やディスク障害等のRASイベントメッセージをアプリケーションで検出し、後処理を組み込むための機能です。（４．２．２　RASイベントハンドリング概要 参照）

（９）RASイベントログ収集・表示機能

ファン異常、筐体内温度異常やディスク障害等の時に、コンソール画面へのRASイベントメッセージ表示とRASイベント情報のログ記録を行います。（７．２　ログファイル収集機能 参照）
（７．３．１　RASイベントログ表示コマンド（raslogpt） 参照）

（10）RAID接続サポート機能　（Dモデルのみ）

RAIDアダプタに接続された内蔵ハードディスクの状態監視／状態表示をサポートします。（２．７　RAID監視機能（Dモデルのみ） 参照）（４．１．１　RAS状態表示コマンド（rasstat）参照）

（11）状態表示デジタルLED機能

本装置のファンや筐体内温度の状態および内蔵ハードディスクの障害予測情報を状態表示デジタルLEDに表示します。このLEDはユーザアプリケーションからも使用できるので、保守員への障害通知などにも使用することができます。

（４．３　状態表示デジタルLED機能 参照）

（12）RASリモート通知機能

ファン異常、筐体内温度異常の検出や内蔵ハードディスクドライブの障害予測検出など装置の状態をSNMP（Simple Network Management Protocol）を利用してリモート通知します。

（４．５　RASリモート通知機能（SNMP） 参照）

このマニュアルでは（２）～（12）の機能について説明します。

その他の機能の詳細については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「７．２　メモリダンプ採取機能」を参照してください。

第２章　RAS機能の監視対象

この章では、RAS機能の監視対象について説明します。

なお、この章で記載しているRAS外部接点インターフェース（HF-BX/LXのみ）のハードウェア仕様や各接点の意味については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」の「４．４．１　コネクタ仕様」および「４．４．２　外部インタフェースケーブル長規定」を参照してください。

２．１　ファン監視機能

ファン監視機能は、この装置に実装されている複数のファンの状態を監視し、いずれかのファンに異常が発生した場合、以下の方法で通知します。

（１）RAS状態表示コマンド（rasstat）

（２）RASイベント情報管理機能

（３）状態表示デジタルLED表示機能

（４）RASリモート通知機能

（５）自動シャットダウン機能

（６）RASライブラリのGetRasStat関数、WaitForRasEvent関数

（１）～（４）については「第４章　ハードウェア状態の確認」を参照してください。

（５）については「５．１　装置の自動シャットダウン」を、（６）については「第６章　ライブラリ関数」を参照してください。

また、RAS外部接点インターフェース（HF-BX/LXのみ）のMCALL接点を使用すれば、RAS外部インターフェースに接続した外部のハードウェアより、この装置の異常を検出することができます。MCALL接点の動作は以下のとおりです。

表２－１　ファン監視における装置の状態とMCALL接点の動作

この装置の状態	MCALL接点
装置起動時 または 電源断時	オープン
ファン正常 または ファン異常から回復	オープン
ファン異常	クローズ

RAS外部接点インターフェース（HF-BX/LXのみ）のMCALL接点はファン監視以外にも使用されているため、ファン異常が回復しても他の監視対象で異常を検出している場合は、オープンしません。

２．２　筐体内温度監視機能

筐体内温度監視機能は、この装置内部の温度センサーにより筐体内温度の状態を監視し、筐体内温度が高温異常になった場合、以下の方法で通知します。

（１）RAS状態表示コマンド（rasstat）
（２）RASイベント情報管理機能
（３）状態表示デジタルLED表示機能
（４）RASリモート通知機能
（５）自動シャットダウン機能
（６）RASライブラリのGetRasStat関数、WaitForRasEvent関数

（１）～（４）については「第４章　ハードウェア状態の確認」を参照してください。
（５）については「５．１　装置の自動シャットダウン」を、（６）については「第６章　ライブラリ関数」を参照してください。

また、RAS外部接点インターフェース（HF-BX/LXのみ）のMCALL接点を使用すれば、RAS外部インターフェース（HF-BX/LXのみ）に接続した外部のハードウェアより、この装置の異常を検出することができます。当該接点の動作は以下のとおりです。

表２－２　筐体内温度監視における装置の状態とMCALL接点の動作

この装置の状態	MCALL接点
装置起動時 または 電源断時	オープン
正常 または 温度異常から回復	オープン
温度異常	クローズ

RAS外部接点インターフェース（HF-BX/LXのみ）のMCALL接点は筐体内温度監視以外にも使用されているため、温度異常が回復しても他の監視対象で異常を検出している場合は、オープンしません。

２．３　ドライブ障害予測機能（SMART監視）

この装置のドライブには障害予測機能（SMART：Self-Monitoring, Analysis and Reporting Technology）が備わっており、常にドライブの動作状態を監視して障害が発生する前にそれを予測することができます。ドライブ障害予測機能は、近い将来ドライブに障害が発生する可能性がある場合、以下の方法で通知します。

（１）RAS状態表示コマンド（rasstat）
（２）RASイベント情報管理機能
（３）状態表示デジタルLED表示機能
（４）RASリモート通知機能
（５）RASライブラリのGetRasStat関数、WaitForRasEvent関数

（１）～（４）については「第４章　ハードウェア状態の確認」を（５）については「第６章　ライブラリ関数」を参照してください。

留意事項

・SMARTは、ドライブのすべての障害を予測することはできません。そのため、SMARTが障害を予測する前に、ドライブが故障する場合もあります。
・ドライブの障害が予測された場合は、速やかにデータのバックアップを取り、ドライブの交換を行ってください。なお、ドライブの交換手順については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「５．４．６　HDDの取り付け／取り外し」を参照してください。
・この機能が監視するのは、OS起動時に認識されているドライブベイ１および２の内蔵ドライブです。ドライブを新規に接続した場合や保守などで交換を行った場合、接続後の初回起動においてドライブの認識に時間がかかり、監視対象として認識されないことがあります。この場合は、この装置を再起動してください。

２．４　ドライブ使用時間監視機能

ドライブ使用時間監視機能は、この装置のドライブ使用時間を積算し、使用時間が既定値を超過した場合、以下の方法で通知します。これにより、ドライブの交換時期を把握し、寿命によるドライブ故障を予防することができます。

（１）RAS状態表示コマンド（rasstat）

（２）RASイベント情報管理機能

（３）RASリモート通知機能

（４）RASライブラリのGetRasStat関数、WaitForRasEvent関数

（１）～（３）については「第４章　ハードウェア状態の確認」を、（４）については「第６章　ライブラリ関数」を参照してください。

留意事項

・この機能が監視するのは、OSが起動してからシャットダウンするまでのOSが動作している時間です。そのため、OSが動作していない場合は使用時間の監視を行いません。

・新規ドライブを実装した場合、当該ドライブの使用時間の積算値は自動でリセットされます。

・この機能は、ドライブ障害を予測するものではありませんが、有寿命品の予防保守の観点から、使用時間が既定値を超過した場合はドライブを交換することを推奨します。なお、ドライブの交換手順については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「５．４．６　HDDの取り付け／取り外し」を参照してください。

・この機能が監視するのは、OS起動時に認識されているドライブベイ１および２の内蔵ドライブです。ドライブを新規に接続した場合や保守などで交換を行った場合、接続後の初回起動においてドライブの認識に時間がかかり、監視対象として認識されないことがあります。この場合は、この装置を再起動してください。

２．５　OSロック監視機能

OSロック監視機能は、この装置に実装されているOS動作監視用タイマを用いて、カーネルの暴走やドライバによるCPU占有などの理由で、高レベル（リアルタイム優先度）のプロセスが動作できない状況（以下、OSロックと称す）が発生した時に状態表示デジタルLEDが「88」の点滅状態になるか、またはシステムがリセットされます。

本機能を有効にするには、３．１　RAS機能設定コマンド（rasconfig）で、自動リトリガ・モード（設定値が4または5）を選択します。

留意事項

・この機能では、リアルタイム優先度のプロセスが動作できない状態をOSロックとして扱います。

・OSロックが発生した場合、OS上のプロセスがスケジュールどおりに動作することができず、処理の遅延が発生することにより、この装置を使用した設備に影響をあたえる恐れがありますので、問題点を速やかに改善してください。

２．６　ウォッチドッグタイマ監視機能

この装置にはウォッチドッグタイマが実装されており、この機能がウォッチドッグタイマを自動的にリトリガすることで、プロセスが正常にスケジューリングされていることを監視します。また、専用のライブラリ関数を使用することで、ユーザプログラムの動作状態監視などに使用することができます。

プロセスが正常にスケジューリングされていない状態になった時には、状態表示デジタルLEDが「88」の点滅状態になるか、またはシステムがリセットされます。

本機能を有効にするには、３．１　RAS機能設定コマンド（rasconfig）で、ユーザアプリの動作状態監視モード（設定値が2または3）を選択します。

２．６．１　ユーザプログラムの動作状態監視に使用する方法

ユーザプログラムの動作状態の監視にウォッチドッグタイマを使用する場合は、例えば、ユーザプログラムによりウォッチドッグタイマを定期的にリトリガ（ウォッチドッグタイマのタイムアウトまでの残り時間を初期値に戻す）して、ウォッチドッグタイマのタイムアウトチェック処理を別のプログラムから行う構成となります。この場合の処理フローを以下に示します。

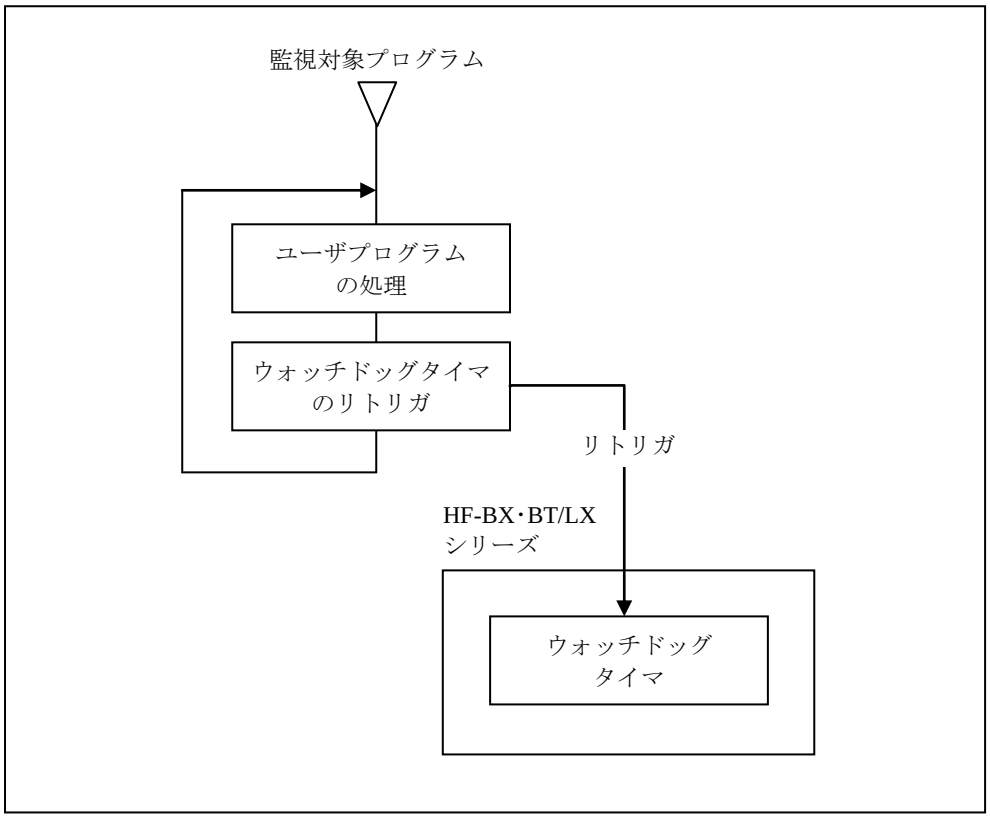

図２－１　ユーザプログラムの動作状態監視処理の例

監視対象プログラムでは、図２－１に示すように定期的にウォッチドッグタイマをリトリガします。

プログラムからウォッチドッグタイマを使用するときは、ライブラリ関数であるWdtControl関数をコールします。WdtControl関数の使用方法については、「６．１．６　ウォッチドッグタイマ制御関数（WdtControl）」を参照してください。

留意事項

WdtControl関数は、Ras機能設定コマンド（rasconfig）のウォッチドッグタイマ設定でユーザアプリケーションの動作状態監視（設定値が2または3）を設定している場合に使用できます。rasconfigコマンドについては、「３．１　RAS機能設定コマンド（rasconfig）」を参照してください。

２．７　RAID監視機能　（Dモデルのみ）

RAIDモデルは、一般にミラーディスク（ミラーリング）として知られているRAID1（以降RAIDと呼びます）の機能を備えています。

RAID監視機能は、この装置におけるRAIDの状態を監視し、RAIDの状態が変化したときに、以下の方法でユーザやアプリケーションに通知します。

（１）RAS状態表示コマンド（rasstat）
（２）RASイベント情報管理機能
（３）状態表示デジタルLED表示機能
（４）RASリモート通知機能
（５）RASライブラリのGetRasStat関数、WaitForRasEvent関数

（１）～（４）については「第４章　ハードウェア状態の確認」を、（５）については「第６章　ライブラリ関数」を参照してください。

また、RAS外部接点インタフェース（HF-BX/LXのみ）のMCALL接点を使用すれば、RAS外部インターフェース（HF-BX/LXのみ）に接続した外部のハードウェアより、この装置の異常を検出することができます。当該接点の動作は以下のとおりです。

表２－３　RAID状態とMCALL接点の動作

RAID状態	MCALL接点
正常	オープン
縮退、不明、故障	クローズ

RAS外部接点インターフェース（HF-BX/LXのみ）のMCALL接点はRAIDの状態監視以外にも使用されているため、RAIDの状態が回復しても他の監視対象で異常を検出している場合は、オープンしません。

２．７．１　RAIDの状態遷移について

表２－４にRAIDの状態とその意味、図２－２に状態遷移を示します。

表２－４　RAID状態とその意味

RAID状態	詳細情報	説明
正常	－	冗長性が保たれ、正常に動作している状態
縮退	－	ドライブが1台故障し、冗長性がくずれた状態
	再構築中	ドライブ交換などにより、RAIDを再構築している状態
故障	－	ドライブが2台故障しているまたはドライブが接続されていない状態
不明	－	RAIDの状態取得に失敗した状態

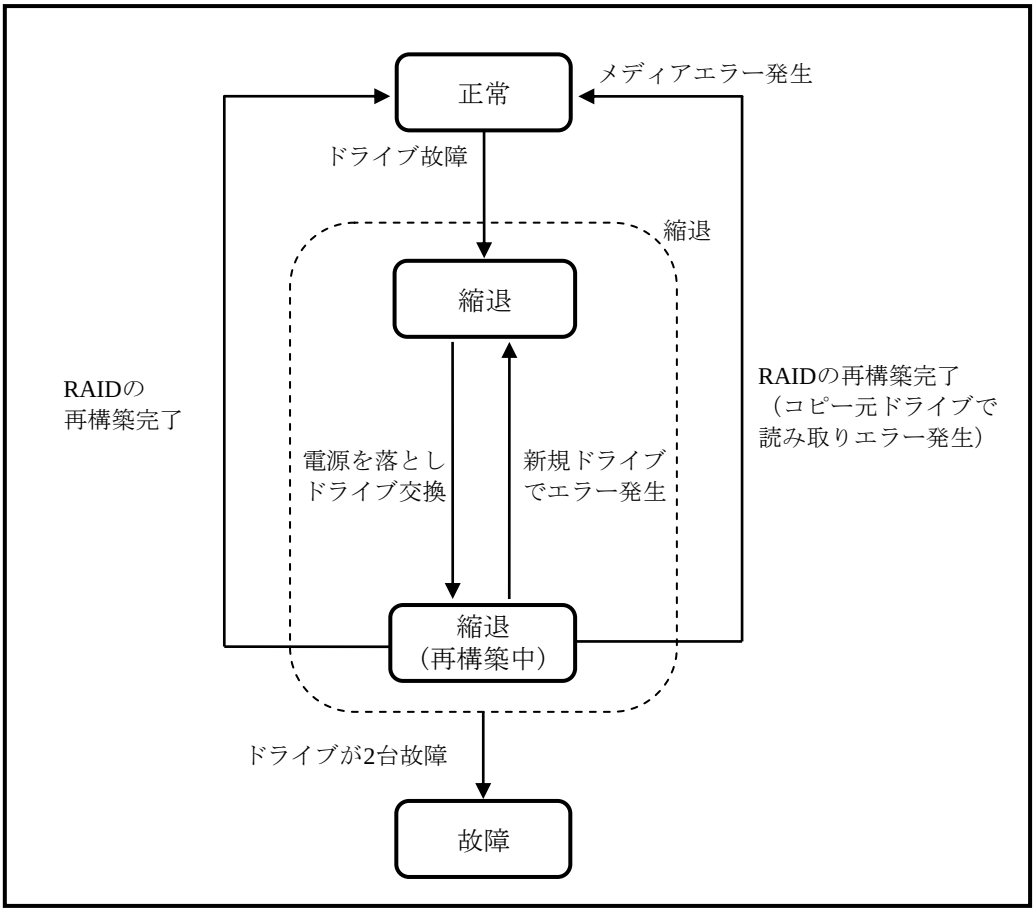

図２－２　RAID状態遷移

２．７．２　メディアエラーについて

メディアエラーとは、RAID状態としては異常が発生していないが、データの健全性に問題がある状態のことです。再構築中にコピー元ドライブからの読み取りエラーが発生した場合、再構築が完了しますが、読み取れなかったセクタのデータは失われており、データの健全性に問題が発生している状態となります。

この他に、RASイベントログ、RAS状態表示コマンドのrasstat 、RASライブラリのGetRasStat関数、WaitForRasEvent関数では、メディアエラー発生の有無を確認できます。

第３章　RAS機能の設定

３．１　RAS機能設定コマンド（rasconfig）

３．１．１　概要

＜名前＞

rasconfig － RAS機能の設定情報表示／設定

＜書式＞

rasconfig　[オプション]

＜実行権限＞

本コマンドは、root権限でのみ実行可能です。

＜機能＞

rasconfigは、RAS機能の設定情報の表示およびRAS機能の設定更新を行うためのコマンドです。

以下の設定ができます。

（１）　ファン異常検出による自動シャットダウン機能の ON/OFF
（２）　温度異常検出による自動シャットダウン機能の ON/OFF
（３）　自動シャットダウン遅延時間の設定
（４）　ウォッチドッグタイマ関連の設定
（５）　状態表示デジタル LED の設定

オプション指定なしでコマンドを実行した場合は、現在のRAS機能設定情報が表示されます。

RAS機能の設定を行う場合は、-eオプションを指定してコマンドを実行します。

このとき、RAS機能の設定を対話形式で行えます。　ウォッチドッグタイマの設定を変更した場合、システムの再起動を促す下記メッセージが出力されます。

> NOTICE: Watchdog Timer function becomes effective after the system reboot.

＜オプション＞

-e　　：　RAS 機能の設定を対話形式で開始します。

＜診断＞

処理が正常に終了した場合、exitステータス0を返します。異常終了した場合はexitステータス0以外を返します。

＜エラーメッセージ＞

表３－１　rasconfigコマンドのエラーメッセージ一覧

No	エラーメッセージ	内容
1	Usage: rasconfig　[-e]	コマンド使用文法に誤りがあります。

＜使用方法＞

RAS機能設定コマンドは、rasconfig -eを実行すると、対話式の問い合わせを行います。
各項目の一番左で表示されている値が現在の値です。設定値を入力せずにEnterキーを押下すると現在の設定値が設定されます。問い合わせに指定できる値以外を入力すると、設定終了の問い合わせを行います。’q’を入力すると、以降の設定をスキップし、設定保存の実行有無確認メッセージが表示されます。

”##### Save configuration? (y/n) : ” では、

’y’：設定値を保存。

’n’：設定値を保存せず終了。

現在の設定値の参照は、rasconfigを実行すると表示します。

＜**rasconfig**コマンドの表示例＞

```
# rasconfig
*** Auto shutdown ***
   FAN abnormal                    : on
   Temperature abnormal            : off
   Shutdown delay time[sec]        : 30
*** Watchdog Timer ***
   Watchdog Timer retrigger        : 1. Not used.
*** DigitalLED setting ***
   Display Hardware Status Code    : on

#
# rasconfig -e
*** Auto shutdown ***
   FAN abnormal (on/off/q)              : on
   Temperature abnormal (off/on/q)      : on
   Shutdown delay time[sec] (30/0-120/q)      : 100
*** Watchdog Timer ***
   Watchdog Timer retrigger (1)
      1. Not used.
      2. Retriggered by application program(Blink digital LED).
      3. Retriggered by application program(Reset).
      4. Automatic retrigger(Blink digital LED).
      5. Automatic retrigger(Reset).
      : 4
*** DigitalLED setting ***
   Display Hardware Status Code (on/off/q)      : on
======== Confirm setting ========
*** Auto shutdown ****
   FAN abnormal                    : on
   Temperature abnormal            : on
   Shutdown delay time[sec]        : 100
*** Watchdog Timer ****
   Watchdog Timer retrigger        : 4.Automatic retrigger(Blink digital LED).
*** DigitalLED setting ***
   Display Hardware Status Code    : on
##### Save configuration? (y/n)    : y

NOTICE: Watchdog Timer function becomes effective after the system reboot.

#
```

＜rasconfigコマンドの使用例＞

```
**** Auto shutdown ****
   FAN abnormal (on/off/q)                :                    ①
   Temperature abnormal (off/on/q)        :                    ②
   Shutdown delay time[sec] (30/0-120/q) :                     ③
**** Watchdog Timer ****
   Watchdog Timer retrigger (1)
     1. Not used.                                               ④
     2. Retriggered by application program(Blink digital LED).  ⑤
     3. Retriggered by application program(Reset).              ⑥
     4. Automatic retrigger(Blink digital LED).                 ⑦
     5. Automatic retrigger(Reset).                             ⑧
     :
*** DigitalLED setting ***
   Display Hardware Status Code    (on/off/q) :                ⑨
```

（１）　自動シャットダウン機能設定（*** Auto shutdown ***）

RAS機能設定では、

●ファン異常時　（①）（デフォルト：on）
●温度異常時　（②）（デフォルト：off）
●自動シャットダウン遅延時間　（③）（デフォルト：30）

のそれぞれの場合について自動シャットダウンを行うか否かの設定ができます。
それぞれの項目がonとなっている場合は、自動シャットダウン機能が行われます。現在の設定を変更する場合は、on/offを入力してください。
自動シャットダウン機能の設定を変更する場合は、下記について注意してください。

（注1）自動シャットダウン遅延時間は、リモートシャットダウン入力（RMTSHTDN）信号を検出した場合には、適用されません。

（２）　ウォッチドッグタイマ設定（*** Watchdog Timer ***）

ウォッチドッグタイマ動作として、

1. ウォッチドッグタイマの停止　（④）
2. ユーザアプリの動作状態監視(状態表示デジタルLEDを点滅）　（⑤）
3. ユーザアプリの動作状態監視(リセット)　（⑥）
4. 自動リトリガ（状態表示デジタルLEDを点滅）　（⑦）
5. 自動リトリガ（リセット）　（⑧）

の内ひとつを選択することができます（デフォルト：1. Not used（ウォッチドッグタイマの停止））。

（３）　状態表示デジタル LED 設定（*** DigitalLED Setting ***）

状態表示デジタルLED設定では、

●ハードウェアステータスコード表示　（⑨）（デフォルト：on）

を設定することができます。状態表示デジタルLEDにハードウェアステータスコードを表示する場合 on を、表示しない場合 off を選択してください。

このページは白紙です。

第４章　ハードウェア状態の確認

この装置では、以下の方法によりハードウェア状態を確認することができます。

（１）rasstatコマンドで確認する

この装置のハードウェア状態を確認することができます。詳細は、「４．１　ハードウェア状態表示機能」をご覧ください。

（２）ユーザアプリケーションで確認する

ユーザアプリケーションからRASライブラリを使用し、RASイベント情報を取得することにより、この装置のハードウェア状態を確認することができます。詳細は、「４．２　RASイベント情報管理機能」をご覧ください。

また、ユーザアプリケーションからRASライブラリを使用することにより、この装置のハードウェア状態を取得することができます。

詳細は、「４．６　RASライブラリ（GetRasStat関数、WaitForRasEvent関数）による状態取得」をご覧ください。

（３）この装置前面の状態表示デジタルLEDで確認する

この装置前面の状態表示デジタルLEDにより、この装置のハードウェアに異常が発生したことをお知らせします。このLEDはユーザアプリケーションからも表示できるので、保守員への障害通知などにも使用することができます。

詳細は、「４．３　状態表示デジタルLED機能」をご覧ください。

（４）リモート環境から確認する

リモート環境からこの装置のハードウェア状態を確認することができます。また、ハードウェア状態に変化があった場合、リモート環境に通知します。

詳細は、「４．５　RASリモート通知機能（SNMP）」をご覧ください。

４．１　ハードウェア状態表示機能

４．１．１　RAS状態表示コマンド（rasstat）

＜名前＞

rasstat　－　RAS機能が管理している装置状態の表示

＜書式＞

rasstat　[オプション]

＜実行権限＞

本コマンドは、全ユーザが実行可能です。

＜機能＞

rasstatは、RAS機能が管理している装置状態を表示するコマンドです。
オプション指定無しで簡易表示、オプション指定有りで詳細表示を実行します。
詳細表示モードでは、コマンド表示に対応する項目を全て表示します。
簡易表示モードでは、コマンド表示に対応する項目を、装置毎に要約して表示します。

[表示方法]は下記のとおり
簡易表示モード　　：　各装置毎に要約した装置状態を表示します。
詳細表示モード　　：　表示可能な装置状態を全て表示します。

＜オプション＞

-v　　：詳細表示モードで本機能を実行します。

＜診断＞

処理が正常に終了した場合、exitステータス0を返します。異常終了した場合はexitステータス0以外を返します。

＜エラーメッセージ＞

なし

＜使用例＞（**D**モデル）

```
#
#rasstat
  FAN                    : Error
  TEMP                   : Error
  RAID                   : Degrade[MediaError]
  DRIVE1                 : Rebuild(12%)
  DRIVE2                 : Overrun
#
#
#rasstat -v
  CPU FAN status         : Normal
  SYS FAN status         : Error
  Temperature status     : Normal(35[degC])
  RAID status            : Degrade
  RAID task              : Rebuild(12%)
  Media error            : Yes
  DRIVE1 status          : Rebuild
  DRIVE1 SMART           : No
  DRIVE1 overrun         : No(1000[h])
  DRIVE2 status          : Online
  DRIVE2 SMART           : Yes
  DRIVE2 overrun         : Yes(99990[h])
  Ras mode               : Online
#
```

表４－１　rasstatコマンドで表示される状態表示項目（詳細表示）

No	部位	項目		状態		備考
		内容	値	内容	値	
1	ファン	CPUファン回転状態	CPU FAN status	正常	Normal	
2				異常	Error	
3		ケースファン回転状態	SYS FAN status	正常	Normal	
4				異常	Error	
5	温度	筐体内温度状態	Temperature status	正常	Normal(xx[degC])	
6				温度異常	Error(xx[degC])	
7				高温異常	Danger(xx[degC])	
8	ドライブ	ドライブ状態	DRIVEn(*1) status	正常	Online	(*1) nは1～2 (*4)使用時間 xxxxxは 未実装などの 場合"---"
9				未実装	Not Connected	
10				オフライン	Offline	
11				リビルド中	Rebuild	
12				不明	Unknown	
13		ドライブ障害予測検知状態	DRIVEn(*1) SMART	正常(未検出)	No	
14				障害予測検出	Yes	
15		ドライブ使用時間超過有無	DRIVEn(*1) overrun	正常	No(xxxxx[h]) (*4)	
16				使用時間超過	Yes(xxxxx[h]) (*4)	
17	RAID	RAID状態	RAID status	正常	Optimal	
18				縮退	Degrade	
19				異常	Fail	
20				不明	Unknown	
21		メディアエラー発生状態	Media error	正常(検知なし)	No	
22				異常(検知)	Yes	
23		実行中タスク状態	RAID task	実行中タスクなし	None	(*2)値に続いて タスク進捗率(**%)を表示。
24				リビルド中	Rebuild(**%)(*2)	
25				上記以外の状態	Unknown	
26	RAS	RAS動作中モード	Ras mode	通常モード	Online	
27				RAS障害ｼﾐｭﾚｰｼｮﾝﾓｰﾄﾞ	Simulation	

RAID関連の表示項目は、Dモデルのみ。

表４－２　rasstatコマンドで表示される状態表示項目（簡易表示）

No	装置	項目		状態		要約内容	備考
		内容	値	内容	値		
1	ファン	ファン状態	FAN	正常	Normal	CPUファン、ケースファンの状態を要約。 いずれか１つでも異常がある場合は異常を表示する。	
2				異常	Error		
3	温度	温度状態	TEMP	正常	Normal	温度異常、高温異常を要約。 いずれかの異常がある場合は異常を表示する。	
4				異常	Error		
5	ドライブ	ドライブ状態	DRIVEn(*1)	①　正常	Online	ドライブ状態、ドライブ障害予兆検知状態を要約。 ・①：ドライブ正常状態、障害予兆未検知正常状態、使用時間超過正常状態 ・②～⑤：ドライブ状態 ・⑥：ドライブ障害予兆検知状態 ・⑦：ドライブ使用時間超過状態 ドライブ状態、ドライブ障害予兆検知状態、ドライブ使用時間超過状態のいずれにおいても異常がある場合、以下にて優先表示する。 （１）ドライブ状態＞（２）障害予兆検知状態＞（３）使用時間超過状態	(*1) nは1～2。 (*2)値に続いてRAIDタスク 進捗率(**%) を表示
6				②　未実装	Not Connected		
7				③　RAID(アレイ)構成から切離され未使用	Offline		
8				④　RAID(アレイ)リビルド中	Rebuild(**%)(*2)		
9				⑤　ハードディスク状態認識不可	Unknown		
10				⑥　障害予兆検出	SMART		
11				⑦　使用時間超過	Overrun		
12	RAID	RAID状態	RAID	①　正常	Optimal	RAID状態、メディアエラー発生状態、タスク進捗状況を要約。 以下の書式にて表示する。 書式　RAID状態[メディアエラー検知] ・①～④：RAID状態(常に表示) ・⑤：メディアエラー発生状態(エラー検知時のみ表示) ・⑥：タスク進捗率(実行中タスクがある場合のみドライブ状態に表示)	
13				②　縮退	Degrade		
14				③　異常	Fail		
15				④　RAID状態認識不可	Unknown		
16				⑤　メディアエラー検知	Media Error		
17				⑥　タスク進捗率(%)	0～100(整数)		

RAID 関連の表示項目は、D モデルのみ。

４．２　RASイベント情報管理機能

４．２．１　概要

この機能は、ハードウェア状態異常などのユーザに報告すべきイベントが発生した場合、ユーザアプリケーションに通知します。イベントはログファイルへ保存します。

ユーザアプリケーションは、RASライブラリを使用することで、ハードウェア状態異常などのイベントを取得することができます。

また、raslogptコマンドを使用して、RASイベントログ情報を表示することができます。

４．２．２　RASイベントハンドリング概要

（１）RASイベント情報取得方法

以下の方法によりイベントの発生を検出することができます。

① WaitForRasEvent関数を使用して、イベント情報を取得します。

（６．１．７　RASイベント情報取得関数（WaitForRasEvent）参照）

（２）取得できるRASイベント情報

- ファン異常発生／回復
- 筐体内温度異常発生／回復
- RAID状態の異常発生／回復
- ドライブの異常発生／回復
- ドライブの障害発生を予測
- ドライブの使用時間超過
- リモートシャットダウン信号検知
- 自動シャットダウン開始
- 緊急シャットダウン開始

４．２．３　RASイベントログ情報保存

RASイベント発生時にイベント情報を収集して、RASイベントログファイルに保存します。RASイベントログ内容については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX　取扱説明書」の「６．３　イベントメッセージ」を参照ください。

保存したRASイベント情報は、raslogptコマンドを使用して、情報を表示できます。

（７．３．１　RASイベントログ表示コマンド（raslogpt）参照）

表４－３　RASイベントログファイル仕様

No	項目	仕様
1	ログファイル名	/var/log/rasutl/rasevent.log
2	RASイベント情報保存	件数（512件）
3	保存件数超過時の動作仕様	サイクリック（最古のイベント情報削除）
4	ログファイル形式	バイナリ形式

４．３　状態表示デジタルLED機能

４．３．１　概要

この機能は、ハードウェア状態異常などのユーザに報告すべきイベントが発生した場合、この装置前面に実装している状態表示デジタルLEDにステータスコードを表示することでユーザに通知します。これにより、ハードウェア状態異常などのイベントの発生を知ることができます。

また、ライブラリ関数を使用することにより、ユーザアプリケーションから任意のステータスコードを表示することができます。LED表示は、2桁の16進数として表示されます。

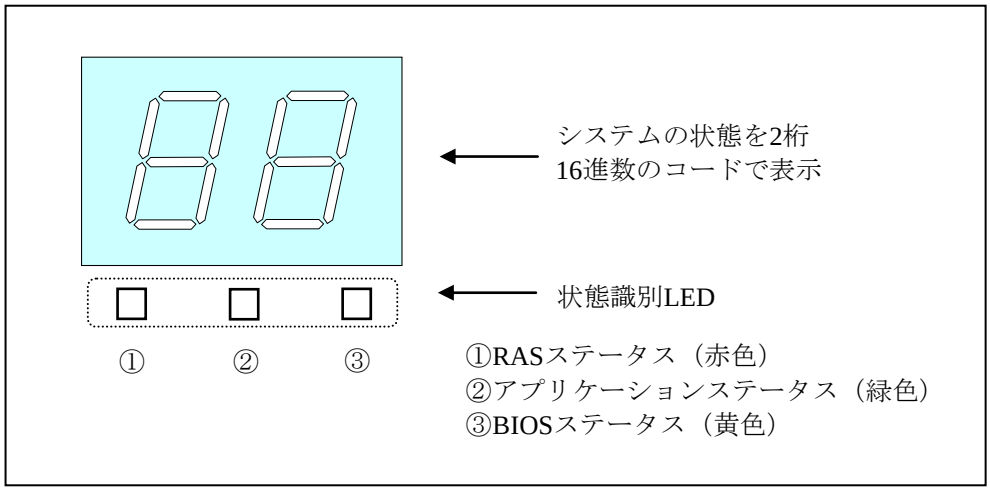

図４－１　状態表示デジタルLED

４．３．２　表示するステータスコード

（１）ハードウェアステータスコード

この装置のハードウェア状態に異常が発生した場合に表示するステータスコードです。
ハードウェア状態が正常な場合は何も表示されません。
具体的には、以下の場合にハードウェアステータスコードを表示します。

- ファン異常発生
- 筐体内温度異常発生
- ドライブの障害発生を予測

ハードウェアステータスコード表示時は、状態識別LEDの左端が点灯します。
ハードウェアステータスコードの内容については、「表４－４　ハードウェアステータスコード一覧」を参照してください。

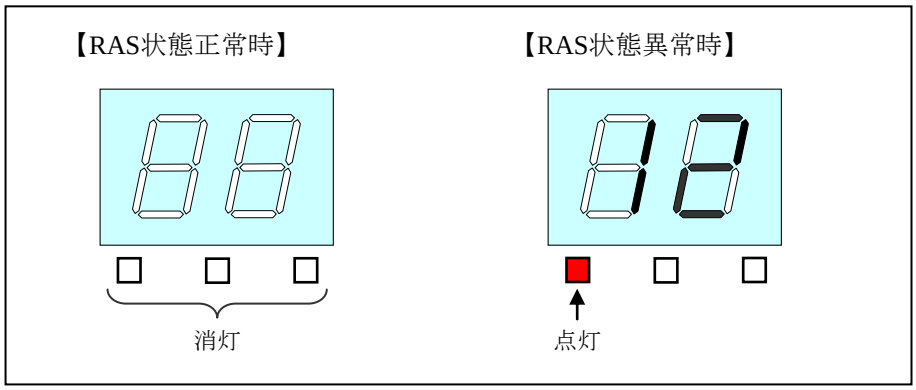

図４－２　ハードウェアステータスコード

表４－４　ハードウェアステータスコード一覧

(○=ｻﾎﾟｰﾄ/×=非ｻﾎﾟｰﾄ)

No	ステータスコード値	監視部位	異常発生詳細	表示対応ﾓﾃﾞﾙ		備考
				A ﾓﾃﾞﾙ	D ﾓﾃﾞﾙ	
1	11	内蔵ファン	ケースファンの異常を検出	→	○	
2	12		CPU ファンの異常を検出	→	○	
3	21	筐体内温度	温度異常を検出	→	○	
4	31	内蔵ドライブ SMART	ドライブベイ 1 のドライブで SMART を検出	→	○	
5	32		ドライブベイ 2 のドライブで SMART を検出	→	○	
6	41	内蔵 RAID	ドライブベイ 1 のドライブが非オンライン	×	○	(*3)
7	42		ドライブベイ 2 のドライブが非オンライン	×	○	(*3)
8	4B		RAID アレイの故障(Fail)を検出	×	○	(*1)
9	4C		RAID アレイの不明(Unknown)を検出	×	○	
10	4D		RAID メディアエラーを検出	×	○	(*2)

(*1)RAIDアレイ故障状態ではシステムが正常に動作出来ないため、本ステータスコード値を表示出来ない場合があります。

(*2)RAIDメディアエラー通知設定(rasraidctl)を有効設定にしている場合のみ表示(製品出荷初期設定は無効設定)

(*3)ドライブ状態の「非オンライン」とは、『オフライン／未実装／リビルド中』のいずれかの状態とします。

（２）アプリケーションステータスコード

この機能が提供するライブラリ関数を使用することにより、ユーザアプリケーションから表示するステータスコードです。

アプリケーションのステータスコード表示時は、状態識別LEDの中央が点灯します。

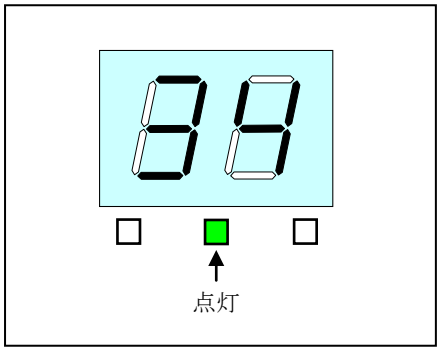

図４－３　アプリケーションステータスコード

（３）STOPエラーコード

システム稼働中に何らかの要因によりSTOPエラーが発生した場合に表示するエラーコードです。STOPエラーコードとして「80」を優先的に表示します。

STOPエラーコード表示時は、状態識別LEDのすべてが点灯します。

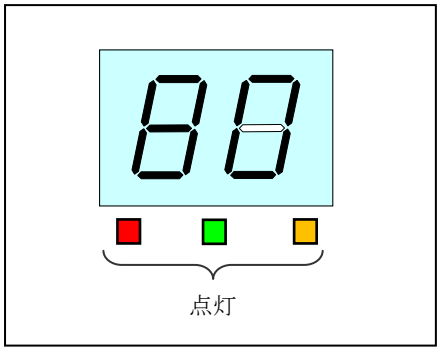

図４－４　STOPエラーコード

（４）ウォッチドッグタイマタイムアウトコード

RAS機能設定コマンド(rasconfig)のウォッチドッグタイマ設定で「ユーザアプリの動作状態監視（状態表示デジタルLEDを点滅）」または「自動リトリガ（状態表示デジタルLEDを点滅）」を設定し、ウォッチドッグタイマ(WDT)のリトリガを行うプロセスが動作不能状態となった場合に表示するエラーコードです。ウォッチドッグタイマタイムアウトコードとして「88」を点滅表示します。

ウォッチドッグタイマタイムアウトコード表示時は、ウォッチドッグタイマタイムアウトが発生する直前の状態識別LEDで点灯または消灯します。

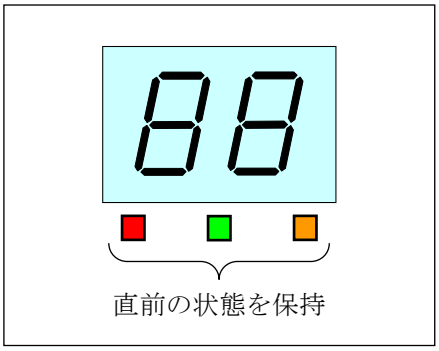

図４－５　ウォッチドッグタイマタイムアウトコード

４．３．３　ステータスコードの表示

（1）ステータスコードの表示タイミング

下記に示すステータスコード表示条件が発生したときに、ステータスコードの表示が行われます。

ステータスコードの表示は、当該コードの表示要因が排除されるか、装置電源OFFされるまで継続されます。主電源OFFまたは電源スイッチONにより表示をリセットします。

表　４－５　ステータスコード表示条件一覧

No	ステータスコード種別	ステータスコード表示条件
1	ハードウェアステータスコード	SMART 監視機能異常検出時
		RAID 監視機能異常検出時
		温度監視機能異常検出時
		ファン監視機能異常検出時
2	アプリケーションステータスコード	RAS ライブラリ関数(SetStCode7seg)使用時
3	STOP エラーコード	カーネルパニック等のメモリダンプ採取要因発生時
4	ウォッチドッグタイマタイムアウトコード	ウォッチドッグタイマタイムアウト発生時(*1)

(*1) RAS機能設定コマンド(rasconfig)のウォッチドッグタイマ設定で
「ユーザアプリの動作状態監視（状態表示デジタルLEDを点滅）」または
「自動リトリガ（状態表示デジタルLEDを点滅）」を設定している場合のみ表示します。

（2）ステータスコードの表示種別一覧

状態識別LED点灯パターンによるステータスコードの表示種別は下記となります。

表　４－６　ステータスコードの表示種別一覧

No	状態識別LED			ステータスコード種別	表示元
	RAS	アプリ	BIOS		
1			●	POST コード	BIOS
2		●	●	(未使用)	－
3	●		●	BIOS エラーコード	BIOS
4		●		アプリケーションステータスコード	ユーザアプリ
5	●	●		(未使用)	－
6	●			ハードウェアステータスコード	RAS 機能
7	●	●	●	STOP エラーコード	RAS 機能

４．３．４　状態表示デジタルLED制御関数

状態表示デジタルLEDを制御するライブラリ関数として、表４－７に示す関数を提供しています。

表４－７　状態表示デジタルLED制御関数一覧

No.	関数名称	機能	共有ライブラリ
1	SetStCode7seg	アプリケーションステータスコードを表示する。	libRASUTL.so
2	TurnOff7seg	アプリケーションステータスコードを非表示にする。	
3	SetMode7seg	ステータス表示モードを設定する。	

上記関数は、RASライブラリ（libRASUTL.so）にて提供します。

ライブラリ関数のインタフェースについては、「６．１．４　状態表示デジタルLED制御関数（SetStCode7seg 、TurnOff7seg、SetMode7seg）」を参照してください。

４．４　内蔵ドライブ監視機能

内蔵ドライブ監視機能は、内蔵ドライブの監視を行い、近い将来にドライブで障害が発生する可能性を予測する機能です。

表　４－８　内蔵ドライブ監視機能概略仕様

No	項目	説明
1	ドライブ障害予測監視機能	内蔵ドライブの監視(SMART)を行い、近い将来にドライブで障害が発生する可能性を予測するための機能です。
2	ドライブ使用時間監視機能	内蔵ドライブの使用時間(OSが起動してからシャットダウンするまでの時間の積算)を監視して、使用時間が既定値を超過した場合に通知するための機能です。

４．４．１　ドライブ障害予測監視機能

ドライブ障害予測監視機能は、内蔵ドライブの監視を一定周期で行い、近い将来にドライブで障害が発生する可能性を予測する機能です。障害予測が検出された場合は各種通知を行います。

HF-BX･BT/LXシリーズの内蔵ドライブに搭載されている自己分析機能(SMART：Self-Monitoring, Analysis and Reporting Technology)を使用し、この機能が保有しているSMART情報を監視することによってドライブの障害予測を行います。

（１）障害予測監視対応のドライブ

本機能で監視対象のドライブ一覧を以下に示します。

表　４－９　障害予測監視対象ドライブ一覧

No	項目	説明
1	ドライブ1	ドライブベイ1に搭載されたHDD
2	ドライブ2	ドライブベイ２に搭載されたHDD

（２）障害予測検出時の動作仕様

RASソフトが一定周期で各ドライブのSMART情報の収集を行い、障害発生の予測検出時に各種通知を実施します。

表　４－１０　HDD障害予測監視機能の動作仕様

No	項目	仕様
1	ドライブ障害予測監視周期	60 分
2	ドライブ障害予測検出時の動作	ログ出力
		ライブラリ関数出力
		SNMP トラップ出力
		状態表示デジタル LED 表示出力

（３）障害予測監視の実行条件

RAID(アレイ)構成ドライブは、ドライブ障害予測監視を行う際にドライブ状態の確認を行い、監視の実行を判断します。

ドライブ状態が正常でない場合は、ドライブへの障害予測監視を無効とします。ドライブ状態が正常の時だけ障害予測監視を有効にして実施します。

表　４－１１　ドライブ状態毎の障害予測監視

No	監視実行の判断	HDD状態	監視有無
1	一定周期 (60 分)	正常	● (有効)
2		オフライン	× (無効)
3		リビルド中	× (無効)
4		不明	× (無効)

４．４．２　ドライブ使用時間監視機能

ドライブ使用時間監視機能は、HF-BX・BT/LXシリーズに内蔵されているドライブの使用時間(OSが起動してからシャットダウンするまでの時間の積算)を監視して、使用時間が既定値を超過した場合に通知を行う機能です。

使用時間の積算はRASソフトの内部カウンタを一定周期で更新することにより行います。

ドライブが交換された場合は、積算時間を初期値（0）に戻します。ドライブが交換されたかどうかは、ドライブ内のシリアル番号を見て、シリアル番号が変わった場合にドライブが交換されたと判断します。

（１）使用時間監視対応のドライブ

本機能で監視対象のドライブ一覧を以下に示します。

表　４－１２　ドライブ使用時間監視対象ドライブ一覧

No	項目	説明
1	ドライブ1	ドライブベイ１に搭載されたHDD
2	ドライブ2	ドライブベイ２に搭載されたHDD

（２）既定時間超過時の動作仕様

RASソフトがドライブ使用時間の内部カウンタを一定周期で更新し、ドライブ使用時間が既定時間を超過した時に各種通知を実施します。

表　４－１３　ドライブ使用時間監視機能の動作仕様

No	項目	仕様
1	ドライブ使用時間更新周期	60 秒
2	ドライブ使用時間超過閾値	20,000 時間
3	既定の使用時間超過時の動作	ログ出力
		ライブラリ関数出力
		SNMP トラップ出力

（３）使用時間の更新条件

ドライブ使用時間の内部カウンタは、対象ドライブの状態が通電状態の時に更新を行い、オフラインや未実装状態(=ドライブが電源断状態)の時には更新しません。

表　４－１４　ドライブ状態毎の使用時間更新条件

No	HDD状態	更新
1	正常	●　（更新実施）
2	オフライン	×　（更新なし）
3	リビルド中	●　（更新実施）
4	不明	×　（更新なし）

４．５　RASリモート通知機能（SNMP）

４．５．１　概要

RASリモート通知機能（SNMP）は、HF-BX・BT/LXシリーズのハードウェア状態を、ネットワークを介したリモート環境から確認できるようにするための機能です。

本機能では、ネットワーク管理用プロトコルとして標準的なI/FであるSNMP（Simple Network Management Protocol）を使用するため、SNMPを採用している多様なネットワーク管理ソフトウェアによるリモート監視を可能としています。また、本機能で対応するSNMPバージョンとして、SNMPv1およびSNMPv2をサポートします。

RASリモート通知機能ではHF-BX・BT/LXシリーズ特有の情報であるRAS情報を公開するため、専用の拡張MIBを定義しています。本機能を使用する場合、この拡張MIBをリモート環境にローディングする必要があります。

拡張MIBの情報に関しては「 ４．５．２　SNMP拡張MIB一覧 」を参照ください。

表４－１５　RASリモート通知機能（SNMP）概略仕様

No	項目	説明
1	RAS 状態照会機能	リモート環境からの要求により、HF-BX・BT/LX シリーズのハードウェアの各状態を通知します。
2	イベント通知機能	RAS ソフトウェアが HF-BX・BT/LX シリーズ本体の障害発生、障害復旧等を検出した場合に、HF-BX・BT/LX シリーズ本体からリモート環境に対してトラップ通知を行います。

留意事項

・RASリモート通知機能が使用するSNMPは、TCP/IPのアプリケーション層プロトコルであり、トランスポート層ではUDP（User Datagram Protocol）を使用しています。
このため、ネットワークの負荷によってはハードウェア状態を正常に受信できない場合があります。

・RASリモート通知機能について、本製品の動作環境におけるすべての機器の動作を保障するものではありません。

・以下の情報を取得するとsnmpdの使用メモリが増加する可能性があります。ご使用の前に、事前に十分検証して頂きますようお願い致します。
OID: UCD-SNMP::extTable(1.3.6.1.4.1.2021.8)
/usr/share/snmp/mibs/UCD-SNMP-MIB.txtに定義されています。

４．５．２　SNMP拡張MIB一覧

RASリモート通知機能（SNMP）では、HF-BX/BT1000/LX固有情報であるRAS情報をベンダー拡張用サブツリー下のHitachi〔iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).hitachi(116)〕サブツリー下に定義しています。

Hitachiサブツリー下には、日立のオブジェクトID登録ルールに従い、system(3)サブツリー、systemExMib(5)サブツリーおよびsystemAP(7)サブツリーの直下に**HF-W/LXシリーズ用サブツリーを定義**しています。HF-BX・BT/LXシリーズでもHF-W/LXシリーズ用サブツリーを使用します。

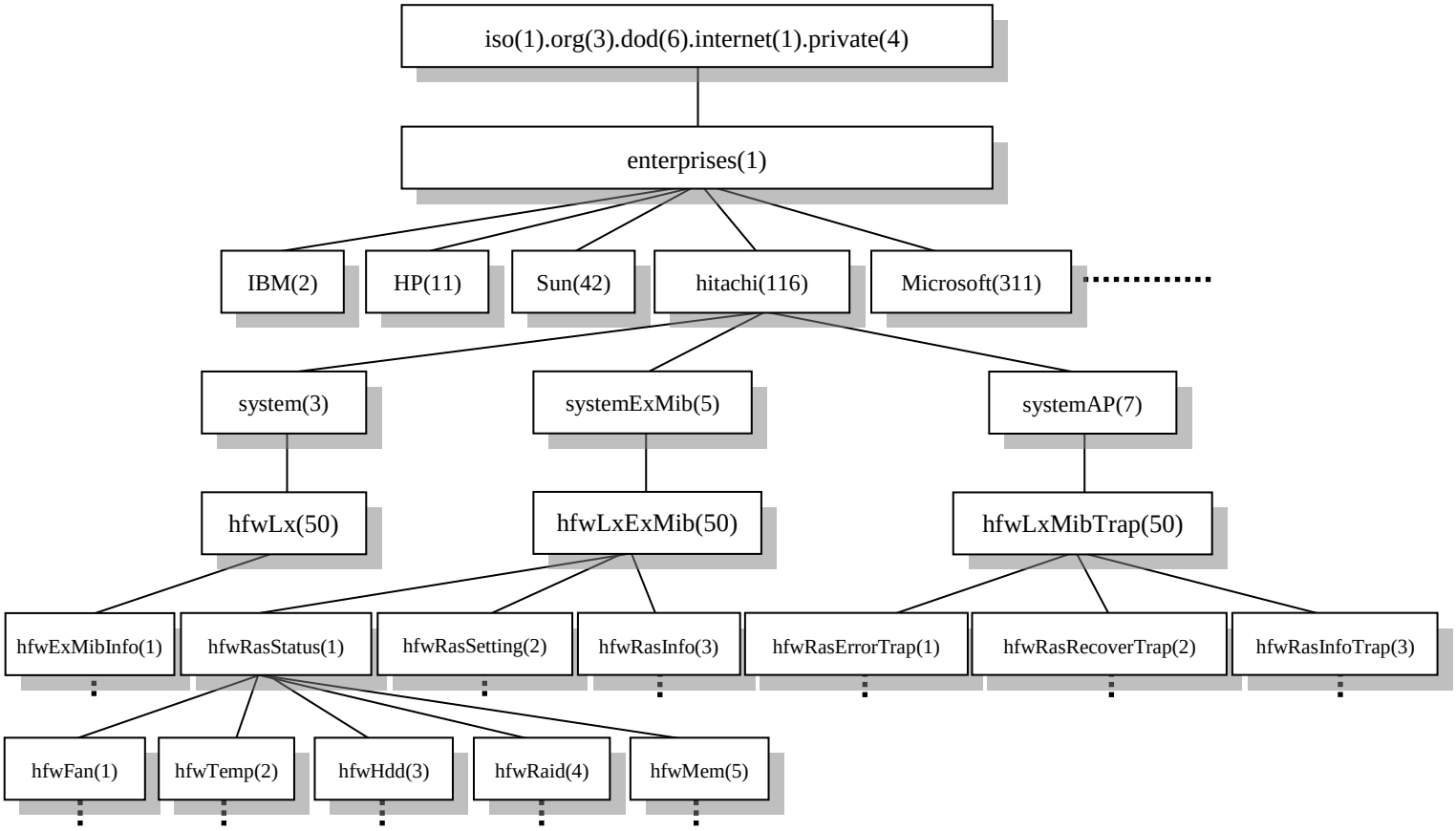

図４－６　サブツリー

４．５．３　hfwLxサブツリー

hfwLx（.1.3.6.1.4.1.116.3.50）下に、hfwExMibInfo(1)サブツリーを定義します。

４．５．４　hfwExMibInfoサブツリー

hfwExMibInfo（.1.3.6.1.4.1.116.3. 50.1）下のサブツリーについて以下に示します。

表　４－１６　hfwExMibInfo下サブツリー

No	識別子	オブジェクト名称	データ型	データ値域	アクセス制限	データ単位	説明
1	1.0	Version	整数	1固定	read-only	－	HF-W/LX用拡張MIBのバージョン情報
2	2.0	Revision	整数	3固定	read-only	－	HF-W/LX用拡張MIBのリビジョン情報

４．５．５　hfwLxExMibサブツリー

hfwLxExMib（.1.3.6.1.4.1.116.5. 50）下に、hfwRasStatus(1)、hfwRasSetting(2)およびhfwRasInfo(3)サブツリーを定義します。

４．５．６　hfwRasStatusサブツリー

hfwRasStatus（.1.3.6.1.4.1.116.5. 50.1）下のサブツリーについて以下に示す。

表　４－１７　hfwRasStatus下サブツリー(1/5)

No	識別子	オブジェクト名称	アクセス制限	データ型	格納データ		説明
					(index)	データ値	
1	1	hfwFan	－	－		－	このサブツリー下ではファン状態に関する情報を定義する。
2	1.1.0	fanNumber	read-only	整数		2	監視対象のファンユニットの個数
3	1.2	fanTable	not-accessible	－		－	－
4	1.2.1	fanEntry	not-accessible	－		－	－
5	1.2.1.1.(index)	fanIndex	read-only	整数	1	1	fanEntryアクセス用のインデックス値(*1)
					2	2	
6	1.2.1.2.(index)	fanName	read-only	文字列	1	CPU fan	ファン識別名称についての情報(*1) “CPU fan” ：CPUファン “SYS fan” ：ケースファン
					2	SYS fan	
7	1.2.1.3.(index)	fanStatus	read-only	整数(列挙型)	1～2	1 (Normal)	ファン回転状態についての情報(*1) 1：正常 2：異常(ファン回転数低下／停止)
						2 (Error)	

表　４－１７　hfwRasStatus下サブツリー(2/5)

No	識別子	オブジェクト名称	アクセス制限	データ型	格納データ		データ単位	説明
					(index)	データ値		
8	2	hfwTemp	－	－		－	－	このサブツリー下では温度状態に関する情報を定義する。
9	2.1.0	tempNumber	read-only	整数		1	－	監視対象の温度ユニットの個数
10	2.2	tempTable	not-accessible	－		－	－	－
11	2.2.1	tempEntry	not-accessible	－		－	－	－
12	2.2.1.1.(index)	tempIndex	read-only	整数	1	1	－	tempEntryアクセス用のインデックス値
13	2.2.1.2.(index)	tempName	read-only	文字列	1	“Internal temperature”	－	温度ユニット識別名称についての情報 “Internal temperature”：筐体内温度
14	2.2.1.3.(index)	tempStatus	read-only	整数(列挙型)	1	1(Normal) 2(Error) 3(Danger)	－	温度状態についての情報 1：正常 2：温度異常(温度異常閾値超過) 3：高温異常(高温異常閾値超過)
15	3	hfwHdd	－	－		－	－	このサブツリー下ではハードディスク状態に関する情報を定義する。
16	3.1.0	hddNumber	read-only	整数		2	－	ハードディスク実装可能台数 2：実装可能台数
17	3.2	hddTable	not-accessible	－		－	－	－
18	3.2.1	hddEntry	not-accessible	－		－	－	－
19	3.2.1.1.(index)	hddIndex	read-only	整数	1 2	1 2	－	hddEntryアクセス用のインデックス値(*2) 1はDRIVE1 , 2はDRIVE2 を表す
20	3.2.1.2.(index)	hddAttribution	read-only	整数(列挙型)	1～2	1 (Signle) 11(RAIDArray)(*4) 99(Unknown)	－	ハードディスク属性(用途)についての情報(*2) 1：シングルディスク用ハードディスク 11：RAIDモデル － アレイ構成用ハードディスク 99：用途情報認識不可

表　４－１７　hfwRasStatus下サブツリー(3/5)

No	識別子	オブジェクト名称	アクセス制限	データ型	格納データ		データ単位	説明
					(index)	データ値		
21	3.2.1.3. (index)	hddStatus	read-only	整数(列挙型)	1～2	1(Online) 2(NotConnected) 11(Offline) 12(Rebuild) 15(Ready)(*4) 99(Unknown)	－	ハードディスク状態についての情報(*2) 1：正常 2：未実装 11：RAID(アレイ)構成から切離され未使用 12：RAID(アレイ)再構築実施中 15：RAID(アレイ)組込み待機中 99：ハードディスク状態認識不可
22	3.2.1.4. (index)	hddPredict	read-only	整数(列挙型)	1～2	1(No) 2(Yes)	－	ハードディスク障害予兆検知状態についての情報(*2) 1：正常(未検出) 2：障害予兆検出
23	3.2.1.5. (index)	hddOverRun	read-only	整数(列挙型)	1～2	1(No) 2(Yes)	－	ハードディスク使用時間超過有無についての情報(*2) 1：正常(使用時間超過なし) 2：使用時間超過
24	3.2.1.6. (index)	hddUseTime	read-only	整数	1～2	0～(UINT32_MAX)	時間(h)	ハードディスク使用時間(*2) （最大値=約490万年）
25	4	hfwRaid	－	—		－	－	このサブツリー下ではRAID状態に関する情報を定義する。
26	4.1.0	raidNumber	read-only	整数		0～1	－	RAIDアレイ数 0：Aモデル 1：Dモデル
27	4.2	raidTable	not-accessible	—		－	－	－
28	4.2.1	raidEntry	not-accessible	—		－	－	－
29	4.2.1.1. (index)	raidIndex	read-only	整数	1	1	－	raidEntryアクセス用のインデックス(Dモデルのみ有効) 1：Dモデル

表　４－１７　hfwRasStatus下サブツリー(4/5)

No	識別子	オブジェクト名称	アクセス制限	データ型	格納データ		データ単位	説明
					(index)	データ値		
30	4.2.1.2.(index)	raidStatus	read-only	整数(列挙型)	1	1(Optimal) 2(CondOptimal) (*4) 3(Degrade) 4(Fail) 99(Unknown)	－	RAID状態についての情報(Dモデルのみ有効) 1：正常 2：条件付正常 3：縮退 4：故障(アレイが壊れ復旧不可能な状態) 99：RAID状態認識不可(未サポートRAIDレベル含む)
31	4.2.1.3.(index)	raidRunningTask	read-only	整数(列挙型)	1	1(None) 2(Rebuid) 3(CopyBack) (*4) 4(ConsistencyCheck) 5(Copy) (*4) 99(Unknown)	－	実行中タスクについての情報(Dモデルのみ有効) 1：実行中タスクなし 2：アレイ再構築中(リビルド) 3：コピーバック中(ホットスペアからのデータ書き戻し) 4：一致化チェック中 5：コピー中 99：タスク状態認識不可
32	4.2.1.4.(index)	raidRunningTaskProgress	read-only	整数	1	0～100	進捗率(%)	タスク進捗状況についての情報(Dモデルのみ有効)
33	4.2.1.5.(index)	raidMediaErrorIndicator	read-only	整数(列挙型)	1	1(No) 2(Yes)	－	メディアエラー発生状態についての情報(Dモデルのみ有効) 1：正常(メディアエラー検知なし) 2：異常(メディアエラー検知)
34	5	hfwMem	－	－	－	－	－	このサブツリー下ではメモリ状態に関する情報を定義する。
35	5.1.0	memNumber	read-only	整数	0	0	－	メモリ実装可能数
36	5.2	memTable	not-accessible	－	－	－	－	－
37	5.2.1	memEntry	not-accessible	－	－	－	－	－

表　４－１７　hfwRasStatus下サブツリー(5/5)

No	識別子	オブジェクト名称	アクセス制限	データ型	格納データ		データ単位	説明
					(index)	データ値		
38	5.2.1.1.(index)	memIndex	read-only	整数	1～2	1～2	―	memEntryアクセス用のインデックス(*4) 1 : DIMM A 2 : DIMM B
39	5.2.1.2.(index)	memName	read-only	文字列	1	“DIMM A”	―	メモリ識別名称についての情報(*4)
					2	“DIMM B”		
40	5.2.1.3.(index)	memStatus	read-only	整数(列挙型)	1～2	1(Normal) 2 (Error) 3(NotMounted)	―	メモリ状態についての情報(*4) 1：正常 2：異常(エラー訂正高頻度発生状態) 3：未実装

*1: index値は、それぞれファンに対応 → 1:CPU fan , 2: SYS fan
*2: index値は、それぞれDRIVEに対応 → 1: DRIVE1 , 2: DRIVE2
*3: index値は、それぞれメモリに対応 → 1:DIMM A , 2:DIMM B
*4: 本機種では使用しないため定義のみとなります。

４．５．７　hfwRasSettingサブツリー

hfwRasSetting(.1.3.6.1.4.1.116.5.50.2）下のサブツリーについて以下に示します。

表　４－１８　hfwRasSetting下サブツリー

No	識別子	オブジェクト名称	データ型	データ値域	アクセス制限	データ単位	説明
1	1.0	hfwFanAutoShutdown	整数(列挙型)	Enable(1) Disable(2)	read-only	－	ファン異常検出時の自動シャットダウン実行有無設定 1：自動シャットダウンを行う 2：自動シャットダウンを行わない
2	2.0	hfwTempAutoShutdown	整数(列挙型)	Enable(1) Disable(2)	read-only	－	温度異常検出時の自動シャットダウン実行有無設定 1：自動シャットダウンを行う 2：自動シャットダウンを行わない
3	3.0	hfwRemoteAutoShutdown	整数(列挙型)	Enable(1) Disable(2)	read-only	－	外部接点[RMTSHTDN]入力検出時の自動シャットダウン実行有無設定 1：自動シャットダウンを行う 2：自動シャットダウンを行わない

４．５．８　hfwRasInfoサブツリー

hfwRasInfo(.1.3.6.1.4.1.116.5.50.3)下のサブツリーについて以下に示します。

表　４－１９　hfwRasInfo下サブツリー

No	識別子	オブジェクト名称	データ型	データ値域	アクセス制限	データ単位	説明
1	1.0	hfwRasMode	整数(列挙型)	Online(1) Simulation(2) Offline(3)	read-only	－	RAS動作中モードについての情報 1：通常モードで動作中 2：RAS障害シミュレーションモードで動作中 3：動作停止

４．５．９　hfwLxMibTrapサブツリー

hfwLxMibTrap(.1.3.6.1.4.1.116.7.50)下に、hfwRasErrorTrap(1)、hfwRasRecoverTrap(2)およびhfwRasInfoTrap(3)サブツリーを定義します。

４．５．１０　hfwRasErrorTrapサブツリー

hfwRasErrorTrap(.1.3.6.1.4.1.116.7.50.1)下のサブツリーについて以下に示します。

表　４－２０　hfwRasErrorTrap下サブツリー

No	識別子	オブジェクト名称	通知データ		説明
			通知オブジェクト	データ値	
1	1	hfwFanError	fanName,	"CPU fan""SYS fan" のいずれか	ファン異常検知の通知
			fanStatus	2 (Error)	
2	2	hfwTempError	tempName,	“Internal temperature”	温度異常検知の通知
			tempStatus	2 (Error) 3 (Danger)	
3	3	hfwSmartDetect	hddIndex	1～2	ハードディスク障害予兆検知の通知 データ値 1はDRIVE1, 2はDRIVE2を表す
			hddPredict	2 (Detected)	
4	4	hfwHddOverRun	hddIndex,	1～2	ハードディスク使用時間超過検知の通知 hddOverRunオブジェクトは、Yes(2)となる。
			hddOverRun	2 (Yes)	
5	5	hfwRaidError	raidIndex,	1	RAID状態異常検知の通知(Dモデルのみ) データ値1は、RAIDのアレイ1となる。
			raidStatus	2 (CondOptima)l (*1) 3(Degrade) 4(Fail) 99(Unknown)	
6	6	hfwMemError	memName,	“DIMM A” 、“DIMM B” のいずれか	エラー訂正高頻度発生検知の通知(*1)
			memStatus	2（Error）	
7	9	hfwMediaError	raidIndex	1	メディアエラー発生検知の通知(Dモデルのみ) データ値1は、RAIDのアレイ1となる。
			raidMediaErrorIndicator	2（Error）	

*1: 本機種では使用しないため定義のみとなります。

４．５．１１　hfwRasRecoverTrapサブツリー

hfwRasRecoverTrap(.1.3.6.1.4.1.116.7.50.2)下のサブツリーについて以下に示します。

表　４－２１　hfwRasRecoverTrap下サブツリー

No	識別子	オブジェクト名称	通知データ		説明
			オブジェクト	データ値	
1	1	hfwFanRecover	fanName,	"CPU fan , "SYS fan" のいずれか	ファン異常からの回復検知の通知
			fanStatus	1 (Normal)	
2	2	hfwTempRecover	tempName,	“Internal temperature”	温度異常からの回復検知の通知
			tempStatus	1 (Normal)	
3	5	hfwRaidRecover	raidIndex	1	RAID状態異常からの回復検知の通知(Dモデルのみ)
			raidStatus	1(Optimal)	
4	6	hfwMemRecover	memName	“DIMM A” , “DIMM B”のいずれか	エラー訂正高頻度発生からの回復検知の通知(*1)
			memStatus	1(Normal)	

*1: 本機種では使用しません。

４．５．１２　hfwRasInfoTrapサブツリー

hfwRasInfoTrap(.1.3.6.1.4.1.116.7.50.3)下のサブツリーについて以下に示します。

表　４－２２　hfwRasInfoTrap下サブツリー

No	識別子	オブジェクト名称	通知オブジェクト(OBJECTS)		説明
			オブジェクト	データ値	
1	1	hfwRasServiceStarted	hfwRasMode	1 (Online)	RASサービス起動の通知
2	2	hfwRasModeChanged	hfwRasMode	1 (Online) 2 (Simulation)	RAS動作モード変更の通知
3	3	hfwRasServiceStopped	hfwRasMode	3 (Offline)	RASサービス停止の通知

４．５．１３　RAS状態照会機能

RAS状態照会機能は、リモート環境からの要求により、HF-BX・BT/LXシリーズハードウェア各状態の監視や制御を行うための機能です。

SNMPによるリモート監視では、監視する側をマネージャ、監視される側をエージェントと呼び、マネージャとエージェント間でコマンドをやり取りすることによって監視と制御を行っています。
マネージャであるリモート環境からエージェントであるHF-BX・BT/LXシリーズ本体に対してMIB情報の取得／設定の要求を発行し、エージェントはその要求を処理して結果をマネージャに応答します。

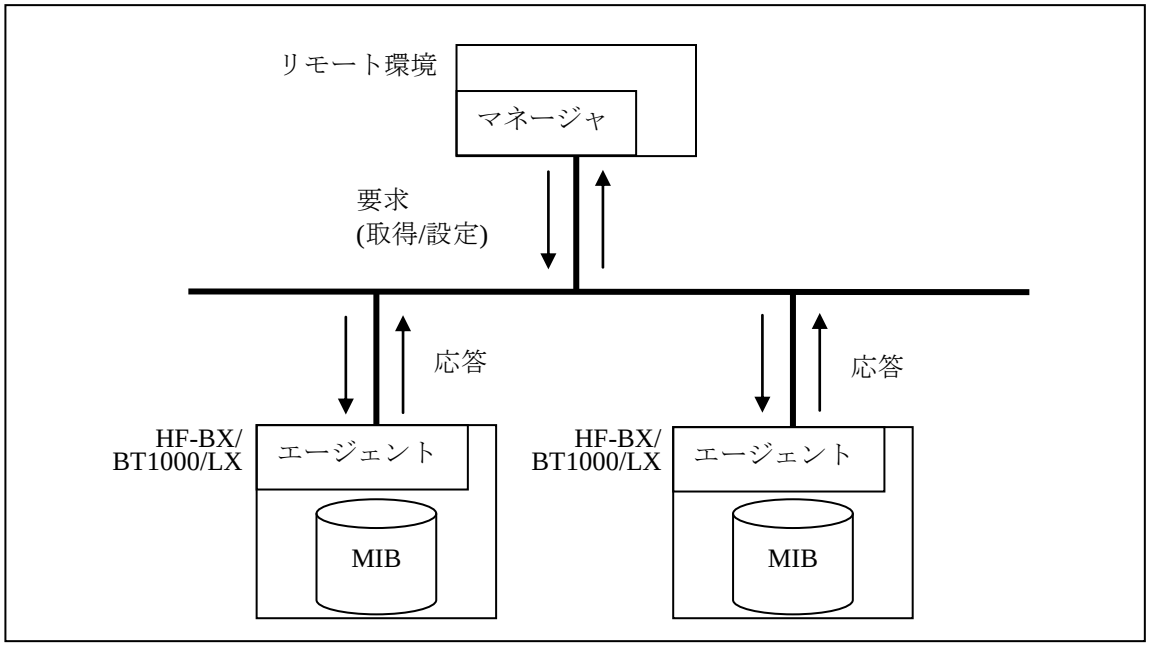

図４－７　状態照会時のネットワーク構成例

本機能で照会できる情報を下記一覧に示します。

表４－２３　照会情報一覧

No	項目	内容
1	HF-BX・BT/LX シリーズ機器識別情報	HF-BX・BT/LX シリーズの機器識別をするための情報。バージョン情報等を保有しています。
2	HF-BX・BT/LX シリーズ情報	HF-BX・BT/LX シリーズのハードウェア各状態の監視や、制御を行うための情報です。

本機能で照会できる拡張MIBとして、HF-BX・BT/LXシリーズ機器識別用MIB情報とHF-BX・BT/LXシリーズ用MIB情報の2つがあります。
HF-BX・BT/LXシリーズ機器識別用MIB情報に関しては ４．５．３　hfwLxサブツリー を参照ください。
HF-BX・BT/LXシリーズ用MIB情報に関しては ４．５．５　hfwLxExMibサブツリー を参照ください。
RAS 状態表示機能で照会できる情報を下記一覧に示します。

表４－２４　RAS状態照会機能の動作仕様

No	項目	内容
1	MIB 情報更新周期	５秒周期にて動作

４．５．１４　イベント通知機能

イベント通知機能は、RASソフトウェアがHF-BX・BT/LXシリーズ本体の障害発生、障害復旧等を検出した場合に、HF-BX・BT/LXシリーズ本体からリモート環境に対してトラップ通知を行う機能です。

エージェントであるHF-BX・BT/LXシリーズ本体の内部でMIB情報を監視し、MIB情報が特定の状態に変化した場合にその旨をマネージャであるリモート環境に対してトラップ通知を行っています。

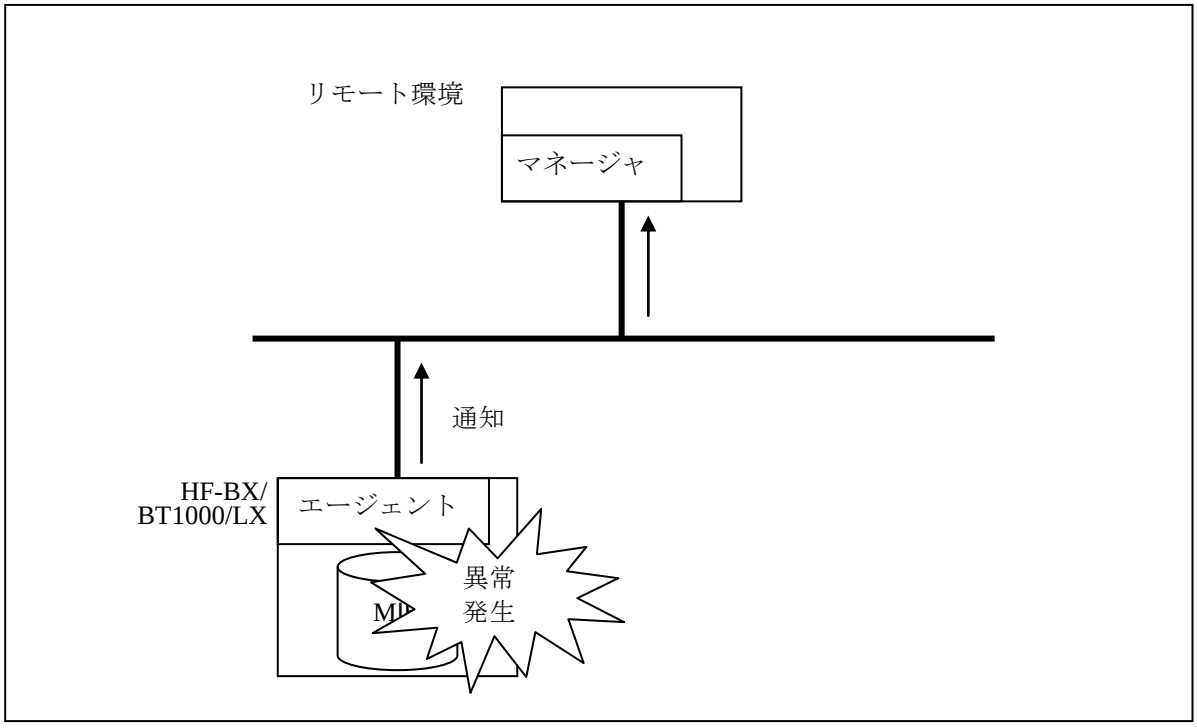

図４－８　イベント通知機能ネットワーク構成例

トラップ通知は下記項目のタイミングで実施します。

表４－２５　トラップ通知一覧

No	項目	内容
1	障害発生時	RAS ソフトウェアが HF-BX・BT/LX シリーズの障害を検出した場合、障害が発生したことをトラップ通知します。
2	障害復旧時	RAS ソフトウェアが HF-BX・BT/LX シリーズの障害の回復を検出した場合、障害が回復したことをトラップ通知します。
3	RAS ソフトウェア起動	RAS ソフトウェア起動時、停止時および RAS 動作モード切替時にその旨をトラップ通知します。

本機能でトラップ通知を使用するための拡張MIBとして、HF-BX・BT/LXシリーズ用トラップMIB情報があります。HF-BX・BT/LXシリーズ用トラップMIB情報に関しては ４．５．９　hfwLxMibTrapサブツリー を参照ください。

４．５．１５　RASリモート通知機能の開始手順

RASリモート通知機能は標準では無効になっているため、本機能を使用する時は以下の手順に従い、RASリモート通知機能を有効にする必要があります。

留意事項

・snmpd.confファイル変更後に必ず**OSの再起動を行い**、snmpd.confファイルの変更内容を反映してください。

（１）　ファイアーウォールで'snmp'を通すよう追加設定を行います

① デスクトップの「システム」をクリック後、「管理」より「ファイアーウォールを設定」をクリックします。

② 「ファイアーウォールの設定」画面左から「その他のポート」を選択し、「追加」ボタンクリック後、ポートとプロトコル画面から順次

161 TCP snmp
161 UDP snmp
162 TCP snmp
162 UDP snmp

を選択して「OK」ボタンをクリックします。
（１ポートずつ追加する処理を４回繰り返します）

③ すべて設定したら画面を閉じてください。

（２）snmpd.conf ファイル定義

/etc/snmp/snmpd.conf ファイルの定義編集を行います。

① コミュニティ情報の登録

/etc/snmp/snmpd.confファイルを編集し、コミュニティ情報の登録を行います。

```
################################################################################
# Access Control
################################################################################

# As shipped, the snmpd demon will only respond to queries on the
# system mib group until this file is replaced or modified for
# security purposes.  Examples are shown below about how to increase the
# level of access.

# By far, the most common question I get about the agent is "why won't
# it work?", when really it should be "how do I configure the agent to
# allow me to access it?"
#
# By default, the agent responds to the "public" community for re
# only access, if run out of the box without any configuration file
# place.  The following examples show you other ways of config
# the agent so that you can change the community names, and giv
# yourself write access to the mib tree as well.
#
# For more information, read the FAQ as well as the snmpd.conf(
# manual page.

####
# First, map the community name "public" into a "security name"

#         sec.name  source          community
#com2sec notConfigUser  default       public
#com2sec hfwlxnet 192.168.1.0/24 hfwlx
#trapcommuity hfwlx
#trap2sink 192.168.1.1 hfwlx
```

１．コミュニティ情報を入力する。
①IPv4アドレス及びマスクビット長を入力
→（例)192.168.1.0/24
※マスクビット長については、リモート接続先固定で使用する場合には指定不要。
②コミュニティ名称を入力
→（例)hfwlx

２．SNMPトラップ通知で使用するためのコミュニティ登録指定を行う。
①コミュニティ名称を入力
※１で指定したコミュニティ名称

３．SNMPトラップ設定を入力する。
①トラップ通知先IPアドレスを指定
②通知コミュニティ名称を入力
※１で指定したコミュニティ名称

（３）snmpd.confの変更内容反映

snmpd.confファイル変更後に**OSの再起動を行い**、snmpd.confファイルの変更内容を反映します。

①マネージャ側（リモート接続先）の設定

RASリモート通知機能を使用するためにはSNMPマネージャソフトにHF-W用の拡張MIBファイルの読込みを行う必要があります。使用するマネージャソフトウェアの手順に従いHF-BX・BT/LXシリーズ用拡張MIBファイルの読込みを行ってください。

HF-BX・BT/LXシリーズ用SNMP拡張MIBファイル名　hfwLxExMib.mib

HF-BX・BT/LXシリーズ用SNMP拡張MIBファイルはRASUTLインストールCDメディアから取得してください。（格納先はDVDドライブ **/mibs/hfwLxExMib.mib**）

４．６　RAS ライブラリ（GetRasStat 関数、WaitForRasEvent 関数）による状態取得

RASライブラリ（GetRasStat関数、WaitForRasEvent関数）を使用することにより、以下に示すRAS機能にて監視しているハードウェア状態を取得することができます。

RASライブラリ関数の詳細については、「６．１．５　RAS状態取得関数（GetRasStat）」、「６．１．７　RASイベント情報取得関数（WaitForRasEvent関数）」を参照してください。

- ファン状態の取得
- 筐体内温度状態の取得
- RAID状態の取得（Dモデルのみ）
- ドライブ状態の取得
- ドライブ障害予測情報の取得
- ドライブ使用時間超過情報の取得

このページは白紙です。

第５章　ハードウェアの制御

この装置では、ハードウェア異常時やリモートシャットダウン接点入力検出時（HF-BX/LXのみ）に、自動的にシャットダウンを行います。

また、RASライブラリ関数によりハードウェア状態を制御することができます。

（１）装置の自動シャットダウン

ハードウェア異常時やリモートシャットダウン接点入力検出時（HF-BX/LXのみ）に、自動的にシャットダウンを行います。「５．１　装置の自動シャットダウン」をご覧ください。

（２）RASライブラリによるハードウェアを制御する

ユーザアプリケーションからRASライブラリを使用することにより、この装置のハードウェアを制御することができます。「５．２　RASライブラリによる制御」をご覧ください。

５．１　装置の自動シャットダウン

この機能は、ファン異常や高温異常など装置を稼働するには危険な状態にある場合に、自動でシャットダウンを行います。これによりプロセッサなどの内蔵部品を熱による劣化より保護し、この装置の誤動作によるシステムの暴走を防止します。また、外部からのリモートシャットダウン信号の入力により自動でシャットダウンを行うこともできます。

５．１．１　ファン異常検出による自動シャットダウン

この装置に実装されているいずれかのファンに異常が発生したことを検出した場合に、自動でシャットダウンします。

・この機能は、rasconfigコマンドで有効／無効を設定することができます。装置出荷時の初期設定は有効です。詳細は、「３．１　RAS機能設定コマンド（rasconfig）」を参照してください。
・この機能によりシャットダウンを行った場合は、OSシャットダウン後に自動的に電源断を行います。
・ファン異常検出を通知するRASイベントを使用して、ユーザアプリケーションからシャットダウンすることもできます。RASイベントについては、「４．２　RASイベント情報管理機能」を参照してください。

留意事項

・ファンの異常が発生した状態でこの装置の動作を継続すると、プロセッサなど内蔵部品の冷却が不充分になり、装置の誤動作によるシステムの暴走や部品の破壊の可能性があります。このため、できるだけこの機能は有効にしてください。
・この機能を使用しない場合でも、ファン異常検出を通知するRASイベントを使用して、ユーザアプリケーションからシャットダウンするようにしてください。
・ファンの交換については、センドバック修理を実施してください。

５．１．２　高温異常検出による自動シャットダウン

この装置内部の温度センサにより筐体内温度が高温異常であることを検出した場合に自動でシャットダウンします。

・この機能は、rasconfigコマンドで有効／無効を設定することができます。装置出荷時の初期設定は無効です。詳細は、「３．１　RAS機能設定コマンド（rasconfig）」を参照してください。
・この機能によりシャットダウンを行った場合は、OSシャットダウン後に自動的に電源断を行います。
・高温異常検出を通知するRASイベントを使用して、ユーザアプリケーションからシャットダウンすることもできます。RASイベントについては、「４．２　RASイベント情報管理機能」を参照してください。

留意事項

・筐体内温度が高いと、熱による部品の極端な劣化が考えられるため、そのままこの装置を稼働状態にすることは機器の寿命の観点からも好ましくありません。しかし、ファン異常が発生していない状態での高温異常は、装置設置場所のエアコンの故障など外部要因によるものと考えられるため、稼働状態のまま高温異常の原因を取り除くことが可能です。このため、装置出荷時における初期設定は無効になっています。
・高温異常発生後もこの装置を稼働状態のままとして筐体内温度が危険なほど高温になってしまった場合は、システムの暴走や部品の破壊を防ぐため、この機能の設定に関わらず強制的にOSシャットダウンを実施して電源断します。

５．１．３　リモートシャットダウン入力検出による自動シャットダウン（HF-BX/LXのみ）

この装置のRAS外部接点インターフェースにおけるリモートシャットダウン接点（RMTSHTDN接点）がクローズしたときに、自動でシャットダウンします。この機能により、離れた場所からリモートでこの装置をシャットダウンすることができます。

・この機能によりシャットダウンを行った場合は、OSシャットダウン後に自動的に電源断を行います。
・リモートシャットダウン入力検出を通知するRASイベントを使用して、ユーザアプリケーションからシャットダウンすることもできます。RASイベントについては、「４．２　RASイベント情報管理機能」を参照してください。

リモートシャットダウン信号は1000ms以上入力してください。

なお、リモートシャットダウン入力は、最短では400msで検出されます。

５．２　RASライブラリによる制御

RASライブラリ関数を使用することにより、ウォッチドッグタイマ、汎用外部接点（HF-BX/LXのみ）や状態表示デジタルLEDを制御することができます。ライブラリ関数の詳細については、「６．１　RASライブラリインターフェース」を参照してください。

- ウォッチドッグタイマの制御…WdtControl関数
- 汎用外部接点出力の制御…GendoControl関数
- 汎用外部接点入力の制御…GetGendi関数
- 状態表示デジタルLEDの制御…SetStCode7seg関数、TurnOff7seg関数、SetMode7seg関数
- RAS機能が管理している装置状態の取得…GetRasStat関数
- RASイベント情報の取得…WaitForRasEvent関数

第６章　ライブラリ関数

ユーザアプリケーションからRASライブラリを使用することで、この装置のハードウェアの状態を取得および制御することができます。

なお、この章で記載しているRAS外部接点インタフェース（HF-BX/LXのみ）のハードウェア仕様や各接点の意味については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「４．４．１　コネクタ仕様」および「４．４．２　外部インタフェースケーブル長規定」を参照してください。

６．１　RASライブラリインタフェース

６．１．１　概要

この章では、RASライブラリが提供する関数のインタフェースについて説明します。

表６－１にRASライブラリ関数の一覧を示します。

以下に示す関数は、RASライブラリ（libRASUTL.so）にて提供され、C言語/C++言語プログラムで使用できます。

本ライブラリ用のヘッダファイルとして「/usr/include/rasutl/rasctl.h」を提供します。

表６－１　RASライブラリ提供関数一覧

No.	関数名称	機能
1	GendoControl	汎用外部接点出力（GENDO0, GENDO1）のオープンやクローズを行います。
2	GetGendi	汎用外部接点入力（GENDI0, GENDI1）の状態を取得します。
3	SetStCode7seg	状態表示デジタルLEDにアプリケーションステータスコードを表示します。
4	TurnOff7seg	状態表示デジタルLEDのアプリケーションステータスコードを非表示にします。
5	SetMode7seg	状態表示デジタルLEDのステータス表示モードを設定します。
6	WdtControl	ウォッチドッグタイマのリトリガを行います。
7	GetRasStat	RAS機能が管理している装置状態の取得を行ないます。
8	WaitForRasEvent	RASイベント情報の取得を行います。

６．１．２　RAS汎用接点の出力制御関数（GendoControl）（HF-BX/LXのみ）

RAS外部接点インタフェースの汎用外部接点への出力をGendoControl関数で制御します。

（１）関数インタフェース（GendoControl）

＜名前＞

GendoControl　－　RAS汎用接点(GENDO0/GENDO1)の信号出力の制御

＜書式＞

```
#include <rasutl/rasctl.h>
int GendoControl( int nPort, int nCmd );
```

＜説明＞

この関数はnCmdで指定した処理を、nPortで指定したRAS外部接点インタフェースの汎用外部接点（GENDO0、GENDO1）に対して行います。

＜オプション＞

[nPort]オプションは下記となります。

RAS_GENDO0_PORT	：	汎用接点(GENDO0)を操作します。
RAS_GENDO1_PORT	：	汎用接点(GENDO1)を操作します。

[nCmd]オプションは下記となります。

RAS_GENDO_OPEN	：	nPortで指定した汎用接点(GENDO0／GENDO1)をオープンします。
RAS_GENDO_CLOSE	：	nPortで指定した汎用接点(GENDO0／GENDO1)をクローズします。

＜戻り値＞

処理が正常に終了した場合は、0を返します。エラーが発生した場合は-1を返し、errnoにエラーを設定します。

＜エラー＞

EINVAL	：	引数 nPort もしくは nCmd が不当です。
EACCES	：	RAS 機能が完全に起動されない（OFFLINE）状態です。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

（２）RAS外部接点インタフェースの汎用外部接点（GENDO0 /GENDO1）の動作

これらの汎用外部接点は、電源ONと電源OFF時はオープン状態です。

図６－１にGendoControl関数を使用したときのGENDO0接点の動作を示します。

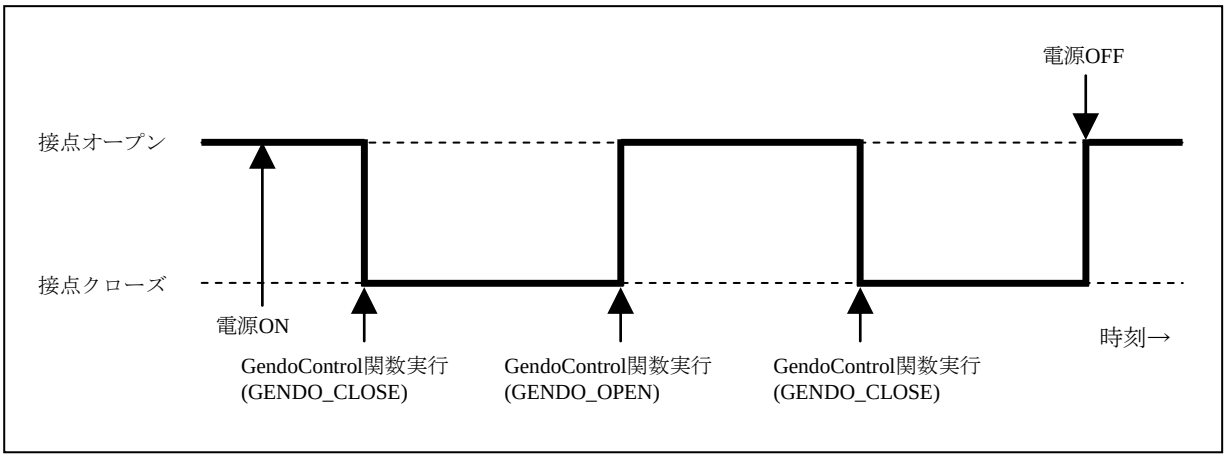

図６－１　GENDO接点の動作（HF-BX/LXのみ）

なお、図６－１の点線は当該接点の各状態を、太線は当該接点の状態の遷移を示します。

※RAS外部接点インタフェースの汎用外部接点（GENDI0、GENDI1）のオープン／クローズ状態は、「図６－１　GENDO接点の動作（HF-BX/LXのみ）」と同様です。

６．１．３　RAS汎用接点の信号入力状態取得関数（GetGendi）（HF-BX/LXのみ）

RAS外部接点インタフェースの汎用外部接点への入力状態をGetGendi関数で取得します。

（１）関数インタフェース（GetGendi）

＜名前＞

GetGendi　－　RAS汎用接点入力（GENDI0,GENDI1）の信号入力状態を取得

＜書式＞

```
#include <rasutl/rasctl.h>
int GetGendi( int nPort );
```

＜説明＞

この関数はnPortで指定したRAS外部接点インタフェースの汎用外部接点（GENDI0、GENDI0）の信号入力状態を取得します。

＜オプション＞

[nPort]オプションは下記となります。

RAS_GENDI0_PORT	：	汎用接点(GENDI0)の状態を取得します。
RAS_GENDI1_PORT	：	汎用接点(GENDI1)の状態を取得します。

＜戻り値＞

処理が正常に終了した場合は、下記いずれかの値を返します。エラーが発生した場合は-1を返し、errnoにエラーを設定します。

RAS_GENDI_OPEN	：	nPortで指定した接点はオープン状態。
RAS_GENDI_CLOSE	：	nPortで指定した接点はクローズ状態。

＜エラー＞

EINVAL	：	引数 nPort が不当です。
EACCES	：	RAS 機能が完全に起動されない（OFFLINE）状態です。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

※RAS 外部接点インタフェースの汎用外部接点（GENDI0 /GENDI1）のオープン／クローズ状態は、「図６－１　GENDO 接点の動作（HF-BX/LX のみ）」と同様です。

６．１．４　状態表示デジタルLED制御関数（SetStCode7seg 、TurnOff7seg、SetMode7seg）

（１）アプリケーションステータスコード表示関数（SetStCode7seg）

＜名前＞

SetStCode7seg　－　状態表示デジタルLEDにアプリケーションステータスコードを表示

＜書式＞

```
#include <rasutl/rasctl.h>
int SetStCode7seg( int nStCode )
```

＜説明＞

SetStCode7segは、状態表示デジタルLEDにアプリケーションステータスコードを表示します。
状態表示デジタルLEDには、nStCodeで指定した値が16進数で表示されます。

＜オプション＞

[nStCode]オプションは下記となります。

0x00 ～ 0xFF	：	表示するアプリケーションステータスコードの値。

＜戻り値＞

成功した場合は0を返します。エラーならば-1を返し、errnoにエラーを設定します。

＜エラー＞

EINVAL	：	指定された nStCode の値が範囲外です。
EACCES	：	RAS 機能が完全に起動されない（OFFLINE）状態です。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

（２）アプリケーションステータスコード非表示関数（TurnOff7seg）

＜名前＞

TurnOff7seg　－　状態表示デジタルLEDに表示されたアプリケーションステータスコードを非表示

＜書式＞

#include <rasutl/rasctl.h>

int TurnOff7seg(void)

＜説明＞

TurnOff7segは、状態表示デジタルLEDに表示されているアプリケーションステータスコードを非表示にします。本関数ではアプリケーションステータスコードの非表示を行うことができますが、ハードウェアステータスコードを非表示にすることはできません。

＜オプション＞

なし。

＜戻り値＞

成功した場合は0を返します。エラーならば-1を返し、errnoにエラーを設定します。

＜エラー＞

EACCES	：	RAS 機能が完全に起動されない（OFFLINE）状態です。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

（３）ステータス表示モード設定関数（SetMode7seg）

＜名前＞

SetMode7seg　－　ステータス表示モード設定

＜書式＞

```
#include <rasutl/rasctl.h>
int SetMode7seg( int nMode )
```

＜説明＞

SetMode7seg()は、状態表示デジタルLEDのステータス表示モードを設定する関数です。
ステータス表示モードを設定するためには、呼び出し元プロセスに適切な権限(CAP_SYS_ADMIN)が必要となります。この関数を呼び出したときに、ステータス表示モードが既にnModeで指定した設定となっている場合(「ハードウェアステータス表示モード」設定状態で「RAS_RASST_MODE」を指定した場合等)、この関数からは処理正常が返ります。

＜オプション＞

[nMode]オプションは下記となります。

RAS_APPST_MODE	：	アプリケーションステータス表示モードに設定します。(ハードウェア異常が発生した場合にハードウェアステータスコードを表示しません。)
RAS_RASST_MODE	：	ハードウェアステータス表示モードに設定します。

＜戻り値＞

成功した場合は0を返します。エラーならば-1を返し、errnoにエラーを設定します。

＜エラー＞

EINVAL	：	引数(nMode)の指定に誤りがあります。
EACCES	：	RAS 機能が完全に起動されない（OFFLINE）状態です。
EPERM	：	呼び出し元プロセスに必要な権限がありません。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

留意事項

本関数では、ハードウェアステータスコードを非表示にすることはできません。

６．１．５　RAS状態取得関数（GetRasStat）

＜名前＞

GetRasStat －　RAS機能が管理している装置状態を取得

＜書式＞

```
#include <rasutl/rasctl.h>
int GetRasStat( struct T_RAS_STAT * ptRasStat )
```

＜説明＞

GetRasStat は、この関数を発行した時点での装置状態が格納された情報を返します。
取得できる状態は、ハードウェア監視機能で監視を行っている監視対象部位の状態（ファン状態、筐体内温度状態、RAID状態、ドライブ状態など）となります。

＜オプション＞

[ptRasStat]オプションは下記となります。

（関数内で情報セット）　：　装置状態情報を格納する領域。関数リターン後、この領域に装置状態情報がセットされています。

格納情報の詳細は＜格納情報詳細＞を参照ください。

＜戻り値＞

成功した場合は0を返します。エラーならば-1を返し、errnoに適切に設定します。

＜エラー＞

EINVAL	:	指定されたptRasStatのアドレスが無効なアドレスです。
EACCES	:	RAS機能が完全に起動されない（OFFLINE）状態です。
E＊＊＊	:	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

＜格納情報詳細＞

①RAS状態を格納する構造体を下記に示します。

```
typedef struct _T_RAS_STAT
{
    // --------------------------------
    int         nRasMode;           : RASソフトウェア動作モード[RAS_STS_MODE_***]
    int         nReserved1[2];      : 予備
    // --------------------------------
    int         nCountOfFan;        : 監視対象ファン数
    struct {
        int     nFanStat;           : ファン状態[RAS_STS_FAN_***]
    } tFAN[4];                      : (16byte)
    // --------------------------------
    int         nCountOfTemp;       : 監視対象温度部位数
    struct {
        int     nTempStat;          : 温度状態[RAS_STS_TEMP_***]
        int     nTempValue;         : 温度値(℃)
    } tTEMP[2];                     : (16byte)
    // --------------------------------
    int         nCountOfMem;        : 監視対象メモリ数
    struct {
        int     nMemStat;           : メモリ状態[RAS_STS_MEM_***]
    } tMEM[4];                      : (16byte)
    // --------------------------------
    int         nCountOfRaid;       : 監視対象RAIDアレイ数
    struct {
        int     nRaidStat;          : RAIDアレイ状態[RAS_STS_RAID_***]+メディアエラー有無
        int     nRaidTask;          : 実行中RAIDタスク種別[RAS_STS_RAID_TASK_***]
        int     nTaskProgress;      : RAIDタスク進捗率(%)
    } tRAID[2];                     : (24byte)
    // --------------------------------
    int         nCountOfDrive;      : 監視対象ドライブ数
    struct {
        int     nDriveStat;         : ドライブ状態[RAS_STS_DRIVE_***]+SMART/使用時間超過有無
        int     nUsedTime;          : ドライブ使用時間(時間)
    } tDRIVE[6];                    : (48byte)
} T_RAS_STAT;
```

②構造体メンバの格納値を表６－２に示します。

表６－２　GetRasStat 構造体メンバ一覧(1/3)

No	構造体メンバ変数				メンバ変数格納情報			ﾓﾃﾞﾙ別情報ｻﾎﾟｰﾄ	
	メンバ変数名		変数型	説明	格納値(define値)		説明	A	D
1	nRasMode		int	RASソフトウェア動作モード	0x5002	(RAS_STS_MODE_ONLINE)	通常モード	→	○
					0x5003	(RAS_STS_MODE_SIMULATION)	障害シミュレーションモード	→	○
2	nCountOfFan		int	監視対象ファン数	2		tFAN[0]：CPUファン tFAN[1]：ケースファン	→	○
3	tFAN[i]	nFanStat	int	ファン状態	0x5101	(RAS_STS_FAN_NORMAL)	ファン状態[正常]	→	○
					0x5102	(RAS_STS_FAN_ERROR)	ファン状態[異常]	→	○
4	nCountOfTemp		int	監視対象温度部位数	1		tTEMP[0]：筐体内温度	→	○
5	tTEMP[i]	nTempStat	int	温度状態	0x5111	(RAS_STS_TEMP_NORMAL)	温度状態[正常]	→	○
					0x5112	(RAS_STS_TEMP_ERROR)	温度状態[異常]	→	○
					0x5113	(RAS_STS_TEMP_DANGER)	温度状態[高温異常]	→	○
		nTempValue	int	温度値(℃)	0～100		温度値(単位：℃)	→	○
6	nCountOfMem		int	監視対象メモリ数	0		tMEM[0]：DIMM-A tMEM[1]：DIMM-B	×	×
7	tMEM[i]	nMemStat	int	メモリ状態	0x5401	(RAS_STS_MEM_NORMAL)	メモリ状態[ビットエラー訂正頻度正常]	×	×
					0x5402	(RAS_STS_MEM_ERROR)	メモリ状態[ビットエラー訂正頻度異常]	×	×
					0x5403	(RAS_STS_MEM_NOTMOUNTED)	メモリ状態[メモリ未実装]	×	×
8	nCountOfRaid		int	監視対象RAIDアレイ数	0		なし(Aモデル)	○	×
					1		tRAID[0]：RAIDアレイ1(Dモデル)	×	○

表６－２　GetRasStat 構造体メンバ一覧(2/3)

No	構造体メンバ変数				メンバ変数格納情報			ﾓﾃﾞﾙ別情報ｻﾎﾟｰﾄ	
	メンバ変数名		変数型	説明	格納値(define値)		説明	A	D
9	tRAID[i]	nRaidStat	int	RAIDアレイ状態	0x5201	(RAS_STS_RAID_OPTIMAL)	RAIDアレイ状態[正常]	×	○
					0x5203	(RAS_STS_RAID_DEGRADE)	RAIDアレイ状態[縮退状態]	×	○
					0x5204	(RAS_STS_RAID_FAIL)	RAIDアレイ状態[異常(復旧不可)]	×	○
					0x5209	(RAS_STS_RAID_UNKNOWN)	RAIDアレイ状態[不明]	×	○
					0x10000	(RAS_STS_RAID_MEDIAERR_DETECTED)	RAIDアレイ状態[メディアエラー検出] (*1)	×	○
		nRaidTask	int	実行中RAIDタスク種別	0x5211	(RAS_STS_RAID_TASK_NONE)	RAIDタスク状態[実行中タスクなし]	×	○
					0x5212	(RAS_STS_RAID_TASK_REBUILD)	RAIDタスク状態[リビルド中]	×	○
					0x5214	(RAS_STS_RAID_TASK_CONSCHECK)	RAIDタスク状態[一致化処理中]	×	○
					0x5219	(RAS_STS_RAID_TASK_UNKNOWN)	RAIDタスク状態[不明]	×	○
		nTaskProgress	int	RAIDタスク進捗率(%)	0～100		RAIDタスク進捗率(単位：%)	×	○
10	nCountOfDrive		int	監視対象ドライブ数	1 2	Aモデル Dモデル	tDRIVE[0]：ドライブ1 tDRIVE[1]：ドライブ2	→	○
11	tDRIVE[i]	nDriveStat	int	ドライブ状態	0x5301	(RAS_STS_DRIVE_ONLINE)	ドライブ状態[正常]	→	○
					0x5302	(RAS_STS_DRIVE_OFFLINE)	ドライブ状態[オフライン]	×	○
					0x5303	(RAS_STS_DRIVE_NOTCONNECTED)	ドライブ状態[未実装]	○	×
					0x5304	(RAS_STS_DRIVE_REBUILD)	ドライブ状態[リビルド中]	×	○
					0x5309	(RAS_STS_DRIVE_UNKNOWN)	ドライブ状態[不明]	×	○

表６－２　GetRasStat 構造体メンバ一覧(3/3)

No	構造体メンバ変数				メンバ変数格納情報			モデル別情報サポート	
	メンバ変数名		変数型	説明	格納値(define値)		説明	A	D
12	tDRIVE[i]	nDriveStat	int	ドライブ状態	0x10000	(RAS_STS_DRIVE_SMART_DETECTED)	ドライブ状態[SMART検出] (*2)	→	○
					0x20000	(RAS_STS_DRIVE_OVERRUN_DETECTED)	ドライブ状態[使用時間閾値超過検出] (*3)	→	○
		nUsedTime	int	ドライブ使用時間(hr)	0～99999		ドライブ使用時間(単位：時間) (ドライブ状態[正常]時有効情報)	→	○

(*1) 本状態検出有無の検査はIS_RAS_RAID_MEDIAERR_DETECTED(nRaidStat)マクロで行えます。また、RAIDアレイ状態[0x52**]の抽出はGET_RAS_RAID_STATE(nRaidStat)マクロで行えます。

(*2) 本状態検出有無の検査はIS_RAS_DRIVE_SMART_DETECTED(nDriveStat)マクロで行えます。また、ドライブ状態[0x53**]の抽出はGET_RAS_DRIVE_STATE(nDriveStat)マクロで行えます。

(*3) 本状態検出有無の検査はIS_RAS_DRIVE_OVERRUN_DETECTED(nDriveStat)マクロで行えます。ドライブ状態[0x53**]の抽出は上記マクロで行えます。

６．１．６　ウォッチドッグタイマ制御関数（WdtControl）

＜名称＞

WdtControl － ウォッチドッグタイマの制御

＜書式＞

```
#include <rasutl/rasctl.h>
int WdtControl( int nCmd, int* pnCount )
```

＜説明＞

この関数は、nCmdで指定した処理をHF-BX・BT/LXシリーズのウォッチドッグタイマに対して行います。

この関数を使用する場合は、RAS機能設定コマンド (rasconfig) の**「ウォッチドッグタイマ機能」の設定を「2または3」に設定する**必要があります。

これ以外のウォッチドッグタイマ設定の場合、この関数はエラーを返します。

＜オプション＞

[nCmd]オプションは下記となります。

RAS_WDT_CMD_SET	：	ウォッチドッグタイマのタイムアウト時間(秒)を設定し、タイムアウト監視を開始します。（タイムアウト監視はワンショットの為、定周期で本オプションを使用してリトリガを行います。）
RAS_WDT_CMD_STOP	：	ウォッチドッグタイマのタイムアウト監視を停止します。

[pnCount]オプションは下記となります。なお、オプション内容は nCmd の指定内容で異なります。

・nCmd が RAS_WDT_CMD_SET の場合

5～63	：	ウォッチドッグタイマのタイムアウト時間。秒単位。

・nCmd が RAS_WDT_CMD_STOP の場合

(未使用)	：	本引数は使用されません。

＜戻り値＞

成功した場合は0を返します。エラーならば-1を返し、errnoにエラーを設定します。

＜エラー＞

EINVAL	：	指定された nCmd の値が範囲外、pnCount の値が不適切。
EPERM	：	ウォッチドッグタイマ機能の動作モードが「2 または 3」以外。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

６．１．７　RASイベント情報取得関数（WaitForRasEvent）

＜名前＞

WaitForRasEvent　－　RASイベント情報の取得

＜書式＞

```
#include <rasutl/rasctl.h>
int WaitForRasEvent ( struct T_RAS_EVENT * ptRasEvent );
```

＜説明＞

WaitForRasEventは、RASイベント情報を取得する関数です。
本関数発行時点で未読のRASイベント情報がある場合、未読のイベント情報を取得することができます。（未読情報は、最大10件分が記録されています）
本関数発行時点で未読のRASイベント情報が無い場合、この関数はRASイベントが発生するまでユーザアプリケーションに処理を戻しません。
取得できる RAS 状態変化情報は、ハードウェア監視機能で監視を行っている監視対象部位の状態（ファン状態、温度状態、RAID 状態、ドライブ状態など）となります。

＜オプション＞

[ptRasEvent]オプションは下記となります。

（関数内で情報セット）　：　RAS イベント情報を格納する領域。
関数リターン後、この領域に RAS イベント情報がセットされています。

格納情報の詳細は＜格納情報詳細＞を参照ください。

＜戻り値＞

処理が正常に終了した場合は、0を返します。エラーが発生した場合は-1を返し、errnoにエラーを設定します。

＜エラー＞

EINVAL	：	指定された ptRasEvent のアドレスが無効なアドレス。
EINTR	：	シグナルを受信。
E＊＊＊	：	ライブラリ関数内で使用しているシステムコールで発生したエラー内容です。

＜格納情報詳細＞

① RASイベント情報を格納する構造体を下記に示します。

```
typedef struct _T_RAS_EVENT
{
    struct timeval      tEvTime;        :  RASイベント発生時刻
    int                 nEvCode;        :  RASイベントコード[RAS_EVT_***]
    int                 nReserved1;     :  予備エリア
    long                nEvParam1;      :  拡張情報1(RASイベントパラメータ1)
    long                nEvParam2;      :  拡張情報2(RASイベントパラメータ2)
    long                nReserved2;     :  予備エリア
} T_RAS_EVENT;
```

② 構造体メンバの格納値を下記に示します。

表６－３　WaitForRasEvent構造体メンバ一覧

No	メンバ	格納値	説明
1	tEvTime	(整数値)	RASイベント発生時刻(相対秒)
2	nEvCode	RAS_EVT_***	RASイベントコード
3	nEvParam1	各イベント毎のパラメータ	
4	nEvParam2	各イベント毎のパラメータ	

③ RASイベントコード毎のパラメータの格納値を下記に示す。

表６－４　パラメータ格納値(1/2)

No	RASイベント情報		RASイベント情報格納構造体メンバ格納値				ﾓﾃﾞﾙ別ｻﾎﾟｰﾄ情報	
	発生部位	イベント内容	<nEvCode>格納値(define値)		<nEvParam1>	<nEvParam2>	A	D
1	ファン	ファン状態[正常]検出	0x6101	(RAS_EVT_FAN_NORMAL)	発生部位(ファン部位)(*1)	－	→	○
2		ファン状態[異常]検出	0x6102	(RAS_EVT_FAN_ERROR)	発生部位(ファン部位)(*1)	－	→	○
3	温度	温度状態[正常]検出	0x6111	(RAS_EVT_TEMP_NORMAL)	発生部位(温度部位)(*2)	－	→	○
4		温度状態[異常]検出	0x6112	(RAS_EVT_TEMP_ERROR)	発生部位(温度部位)(*2)	－	→	○
5		温度状態[高温異常]検出	0x6113	(RAS_EVT_TEMP_DANGER)	発生部位(温度部位)(*2)	－	→	○
6	RAID	RAIDアレイ状態[正常]検出	0x6201	(RAS_EVT_RAID_OPTIMAL)	発生部位(RAIDアレイ部位)(*3)	状態遷移前RAIDアレイ状態(*7)	×	○
7		RAIDアレイ状態[縮退]検出	0x6203	(RAS_EVT_RAID_DEGRADE)	発生部位(RAIDアレイ部位)(*3)	状態遷移前RAIDアレイ状態(*7)	×	○
8		RAIDアレイ状態[故障]検出	0x6204	(RAS_EVT_RAID_FAIL)	発生部位(RAIDアレイ部位)(*3)	状態遷移前RAIDアレイ状態(*7)	×	○

表６－４　パラメータ格納値(2/2)

No	RASイベント情報		RASイベント情報格納構造体メンバ格納値				ﾓﾃﾞﾙ別ｻﾎﾟｰﾄ情報	
	発生部位	イベント内容	<nEvCode>格納値(define値)		<nEvParam1>	<nEvParam2>	A	D
9	RAID	RAIDアレイ状態[不明]検出	0x6209	(RAS_EVT_RAID_UNKNOWN)	発生部位(RAIDアレイ部位)(*3)	状態遷移前RAIDアレイ状態(*7)	×	○
10		RAIDメディアエラー発生検出	0x6222	(RAS_EVT_RAID_MEDIAERR_ERROR)	発生部位(RAIDアレイ部位)(*3)	－	×	○
11	ドライブ	RAID構成ドライブ状態[正常]検出	0x6301	(RAS_EVT_DRIVE_ONLINE)	発生部位(ドライブ部位)(*4)	状態遷移前ドライブ状態(*8)	×	○
12		RAID構成ドライブ状態[オフライン]検出	0x6302	(RAS_EVT_DRIVE_OFFLINE)	発生部位(ドライブ部位)(*4)	状態遷移前ドライブ状態(*8)	×	○
13		RAID構成ドライブ状態[未実装]検出	0x6303	(RAS_EVT_DRIVE_NOTCONNECTED)	発生部位(ドライブ部位)(*4)	状態遷移前ドライブ状態(*8)	×	×
14		RAID構成ドライブ状態[リビルド中]検出	0x6304	(RAS_EVT_DRIVE_REBUILD)	発生部位(ドライブ部位)(*4)	状態遷移前ドライブ状態(*8)	×	○
15		RAID構成ドライブ状態[不明]検出	0x6309	(RAS_EVT_DRIVE_UNKNOWN)	発生部位(ドライブ部位)(*4)	状態遷移前ドライブ状態(*8)	×	○
16		ドライブSMART検知状態検出	0x6312	(RAS_EVT_DRIVE_SMART_ERROR)	発生部位(ドライブ部位)(*4)	－	→	○
17		ドライブ使用時間閾値超過検出	0x6322	(RAS_EVT_DRIVE_OVERRUN_ERROR)	発生部位(ドライブ部位)(*4)	ドライブ使用時間(hr)[100～20000]	→	○
18	外部接点	リモートシャットダウン信号入信検出(BXのみ)	0x6801	(RAS_EVT_RMTSHTDN)	－	－	→	○
19	RASソフト	自動シャットダウン開始	0x6851	(RAS_EVT_SHUT_RUNAUTOSHTDN)	自動ｼｬｯﾄﾀﾞｳﾝ要因ｲﾍﾞﾝﾄ(*5)	ｼｬｯﾄﾀﾞｳﾝ開始までの遅延時間(秒)[0～120]	→	○
20		緊急シャットダウン開始	0x6852	(RAS_EVT_SHUT_RUNEMERGSHTDN)	緊急ｼｬｯﾄﾀﾞｳﾝ要因ｲﾍﾞﾝﾄ(*6)	－	→	○

(*1)発生部位(ファン部位)

0：CPUファン 1：ケースファン

(*2)発生部位(温度部位)

0：筐体内温度

(*3)発生部位(RAIDアレイ部位)

0：RAIDアレイ1(System)

(*4)発生部位(ドライブ部位)

0：ドライブ1 1：ドライブ2

(*5)発生部位(自動ｼｬｯﾄﾀﾞｳﾝ要因ｲﾍﾞﾝﾄ)

0x6102(RAS_EVT_FAN_ERROR)
0x6112(RAS_EVT_TEMP_ERROR)
0x6113(RAS_EVT_TEMP_DANGER)

(*6)発生部位(緊急ｼｬｯﾄﾀﾞｳﾝ要因ｲﾍﾞﾝﾄ)

0x6801(RAS_EVT_RMTSHTDN)

(*7)状態遷移前RAIDアレイ状態

0x5201(RAS_STS_RAID_OPTIMAL)[正常]
0x5203(RAS_STS_RAID_DEGRADE)[縮退]
0x5204(RAS_STS_RAID_FAIL)[故障]
0x5209(RAS_STS_RAID_UNKNOWN)[不明]

(*8)状態遷移前ドライブ状態

0x5301(RAS_STS_DRIVE_ONLINE) ：[正常]
0x5302(RAS_STS_DRIVE_OFFLINE) ：[オフライン]
0x5303(RAS_STS_DRIVE_NOTCONNECTED) ：[未実装]
0x5304(RAS_STS_DRIVE_REBUILD) ：[リビルド中]
0x5309(RAS_STS_DRIVE_UNKNOWN) ：[不明]

６．１．８　サンプルプログラム

RASライブラリとして提供するライブラリ関数の使用方法を示したサンプルコードを提供しています。下表に、提供するサンプルコードファイル一覧を記します。

表６－５　サンプルプログラム一覧

No.	ファイル名称	内容
1	gendo.c	GendoControl()関数の使用方法を示したサンプルコード
2	gendi.c	GetGendi()関数の使用方法を示したサンプルコード
3	7seg.c	SetStCode7seg()/TurnOff7seg()/SetMode7seg()関数の使用方法を示したサンプルコード
4	wdt.c	WdtControl()関数の使用方法を示したサンプルコード
5	rasstat.c	GetRasStat()関数の使用方法を示したサンプルコード
6	rasevent.c	WaitForRasEvent()関数の使用方法を示したサンプルコード

次ページ以降に記載するサンプルコードファイルは、下記RASソフトウェアディレクトリに同梱しています。

/usr/rasutl/examples

各サンプルコードファイルのファイル属性については、一般的なテキストファイル属性「所有者(root)、アクセス権限(0644)」とします。

各サンプルコードのコンパイル方法は、サンプルコードの先頭行に記述しています。

※サンプルプログラムに記載の「¥」コードは、Linux環境では「/」(バックスラッシュ)コードで表示されます。

（１）gendo.c

```
// cc -o gendo gendo.c -lRASUTL
#include <stdio.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	int nPort	= RAS_GENDO0_PORT;
	int nCmd	= RAS_GENDO_OPEN;

	/* set open GENDO0 */
	nPort = RAS_GENDO0_PORT;
	nCmd	= RAS_GENDO_OPEN;
	nResult = GendoControl( nPort, nCmd );
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed GendoControl(GENDO0,OPEN). errno=(%d)¥n", errno );
		return 1;
	}

	/* set close GENDO1 */
	nPort = RAS_GENDO1_PORT;
	nCmd	= RAS_GENDO_CLOSE;
	nResult = GendoControl( nPort, nCmd );
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed GendoControl(GENDO1,CLOSE). errno=(%d)¥n", errno );
		return 1;
	}

	return 0;
}
```

（２）gendi.c

```
// cc -o gendi gendi.c -lRASUTL
#include <stdio.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	int nPort	= RAS_GENDI0_PORT;

	/* get GENDI0 */
	nPort = RAS_GENDI0_PORT;
	nResult = GetGendi( nPort );
	switch ( nResult )
	{
	case RAS_GENDI_OPEN:
		fprintf( stdout, "GetGendi(): GENDI0 --> OPEN¥n" );
		break;
	case RAS_GENDI_CLOSE:
		fprintf( stdout, "GetGendi(): GENDI0 --> CLOSE¥n" );
		 break;
	default:
		fprintf( stderr, "ERROR: failed GetGendi(GENDI0). rtn=(%d), errno=(%d)¥n", nResult, errno );
		return 1;
	}

	return 0;
}
```

（３）7seg.c

```
// cc -o 7seg 7seg.c -lRASUTL
#include <stdio.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	int nMode	= RAS_APPST_MODE;
	int nStCode = 0x00;

	/* set mode */
	nMode = RAS_APPST_MODE;
	nResult = SetMode7seg( nMode );
	if ( nResult == -1 )
	 {
		fprintf( stderr, "ERROR: failed SetMode7seg(APPSTMODE). errno=(%d)¥n", errno );
		return 1;
	}

	/* set 0xff */
	nStCode = 0xff;
	nResult = SetStCode7seg( nStCode );
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed SetStCode7seg(). errno=(%d)¥n", errno );
		return 1;
	}
	sleep( 1 );

	/* turn off led */
	nResult = TurnOff7seg();
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed TurnOff7seg(). errno=(%d)¥n", errno );
		return 1;
	}

	return 0;
}
```

（４）wdt.c

```
// cc -o wdt wdt.c -lRASUTL
#include <stdio.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	int nLoop	= 0;
	int nCmd	= RAS_WDT_CMD_STOP;
	int nCount	= 0;

	/* set watchdog timer */
	for ( nLoop = 0; nLoop < 10; nLoop++ )
	{
		nCmd = RAS_WDT_CMD_SET;
		nCount = 20;
		nResult = WdtControl( nCmd, &nCount );
		if ( nResult == -1 )
		{
			fprintf( stderr, "ERROR: failed WdtControl(SET). errno=(%d)¥n", errno );
			return 1;
		}
		sleep( 5 );
	}

	/* stop watchdog timer */
	nCmd = RAS_WDT_CMD_STOP;
	nResult = WdtControl( nCmd, NULL );
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed WdtControl(STOP). errno=(%d)¥n", errno );
		return 1;
	}

	return 0;
}
```

（５）rasstat.c

```
// cc -o rasstat rasstat.c -lRASUTL
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	int nIndex  = 0;
	T_RAS_STAT tRasStat;

	/* get RAS status */
	memset( &tRasStat, 0, sizeof( tRasStat ) );
	nResult = GetRasStat( &tRasStat );
	if ( nResult == -1 )
	{
		fprintf( stderr, "ERROR: failed GetRasStat(). errno=(%d)¥n", errno );
		return 1;
	}

	fprintf( stdout, "RasMode      = 0x%x(%s)¥n", tRasStat.nRasMode,
			( tRasStat.nRasMode == RAS_STS_MODE_ONLINE ) ? "Online" : "Simulation" );
	for ( nIndex = 0; nIndex < tRasStat.nCountOfFan; nIndex++ )
	{
		switch ( tRasStat.tFAN[nIndex].nFanStat )
		{
		case RAS_STS_FAN_NORMAL:
			fprintf( stdout, "Fan[%d]       = (Normal)¥n", nIndex );
			break;
		case RAS_STS_FAN_ERROR:
			fprintf( stdout, "Fan[%d]       = (Error)¥n", nIndex );
			break;
		default:
			break;
		}
	}
	for ( nIndex = 0; nIndex < tRasStat.nCountOfTemp; nIndex++ )
	{
		switch ( tRasStat.tTEMP[nIndex].nTempStat )
		{
		case RAS_STS_TEMP_NORMAL:
			fprintf( stdout, "Temp[%d]      = (Normal)¥n", nIndex );
			break;
		case RAS_STS_TEMP_ERROR:
			fprintf( stdout, "Temp[%d]      = (Error)¥n", nIndex );
			break;
		case RAS_STS_TEMP_DANGER:
			fprintf( stdout, "Temp[%d]      = (Danger)¥n", nIndex );
			break;
		default:
			break;
		}
		fprintf( stdout, "TempValue[%d] = %d¥n", nIndex, tRasStat.tTEMP[nIndex].nTempValue );
	}
```

(次頁続く)

(前頁続き)

```
    for ( nIndex = 0; nIndex < tRasStat.nCountOfMem; nIndex++ )
    {
        switch ( tRasStat.tMEM[nIndex].nMemStat )
        {
        case RAS_STS_MEM_NORMAL:
            fprintf( stdout, "Mem[%d]       = (Normal)¥n", nIndex );
            break;
        case RAS_STS_MEM_ERROR:
            fprintf( stdout, "Mem[%d]       = (Error)¥n", nIndex );
            break;
        case RAS_STS_MEM_NOTMOUNTED:
            fprintf( stdout, "Mem[%d]       = (Not-Mounted)¥n", nIndex );
            break;
        default:
            break;
        }
    }
    for ( nIndex = 0; nIndex < tRasStat.nCountOfRaid; nIndex++ )
    {
        fprintf( stdout, "Raid[%d]      = (Media error is %d)¥n", nIndex,
                    IS_RAS_RAID_MEDIAERR_DETECTED( tRasStat.tRAID[nIndex].nRaidStat ) );
        switch ( GET_RAS_RAID_STATE( tRasStat.tRAID[nIndex].nRaidStat ) )
        {
        case RAS_STS_RAID_OPTIMAL:
            fprintf( stdout, "                  = (Optimal)¥n" );
            break;
        case RAS_STS_RAID_DEGRADE:
            fprintf( stdout, "                  = (Degrade)¥n" );
            break;
        case RAS_STS_RAID_FAIL:
            fprintf( stdout, "                  = (Fail)¥n" );
            break;
        case RAS_STS_RAID_UNKNOWN:
            fprintf( stdout, "                  = (Unknown)¥n" );
            break;
        default:
            break;
        }
        switch ( tRasStat.tRAID[nIndex].nRaidTask )
        {
        case RAS_STS_RAID_TASK_NONE:
            fprintf( stdout, "RaidTask[%d]    = (None)¥n", nIndex );
            break;
        case RAS_STS_RAID_TASK_REBUILD:
            fprintf( stdout, "RaidTask[%d]    = (Rebuild(%d%))¥n", nIndex,
                                                    tRasStat.tRAID[nIndex].nTaskProgress );
            break;
        case RAS_STS_RAID_TASK_UNKNOWN:
            fprintf( stdout, "RaidTask[%d]    = (Unknown)¥n", nIndex );
            break;
        default:
            break;
        }
    }
```

(前頁続き)

```
    for ( nIndex = 0; nIndex < tRasStat.nCountOfDrive; nIndex++ )
    {
        fprintf( stdout, "Drive[%d]    = (Smart is %d)¥n", nIndex,
                    IS_RAS_DRIVE_SMART_DETECTED( tRasStat.tDRIVE[nIndex].nDriveStat ) );
        fprintf( stdout, "                = (Overrun is %d)¥n",
                    IS_RAS_DRIVE_OVERRUN_DETCTED( tRasStat.tDRIVE[nIndex].nDriveStat ) );
        switch ( GET_RAS_DRIVE_STATE( tRasStat.tDRIVE[nIndex].nDriveStat ) )
        {
        case RAS_STS_DRIVE_ONLINE:
            fprintf( stdout, "                = (Online)¥n" );
            break;
        case RAS_STS_DRIVE_OFFLINE:
            fprintf( stdout, "                = (Offline)¥n" );
            break;
        case RAS_STS_DRIVE_NOTCONNECTED:
            fprintf( stdout, "                = (Not-Connected)¥n" );
            break;
        case RAS_STS_DRIVE_REBUILD:
            fprintf( stdout, "                = (Rebuild)¥n" );
            break;
        case RAS_STS_DRIVE_UNKNOWN:
            fprintf( stdout, "                = (Unknown)¥n" );
            break;
        default:
            break;
        }
         fprintf( stdout, "UsedTime[%d]    = %d¥n", nIndex, tRasStat.tDRIVE[nIndex].nUsedTime );
    }

    return 0;
}
```

（６）rasevent.c

```
// cc -o rasevent rasevent.c -lRASUTL
#include <stdio.h>
#include <string.h>
#include <time.h>
#include <errno.h>
#include <rasutl/rasctl.h>

int main( int argc, char** argv )
{
	int nResult = 0;
	struct tm*	ptTime = NULL;
	T_RAS_EVENT tRasEvent;

	/* get RAS event */
	while ( 1 )
	{
		memset( &tRasEvent, 0, sizeof( tRasEvent ) );
		nResult = WaitForRasEvent( &tRasEvent );
		if ( nResult == -1 )
		{
			if ( errno == EINTR )
			{
				continue;
			}
			fprintf( stderr, "ERROR: failed WaitForRasEvent(). errno=(%d)¥n", errno );
			return 1;
		}
		ptTime = localtime( &tRasEvent.tEvTime.tv_sec );
		fprintf( stdout, "WaitForRasEvent(). "
					"[%04d/%02d/%02d %02d:%02d:%02d.%03d]:"
					"event=[0x%x],param1=[0x%x],param2=[0x%x]¥n",
				(ptTime->tm_year+1900),
				(ptTime->tm_mon+1), ptTime->tm_mday, ptTime->tm_hour, ptTime->tm_min,
				ptTime->tm_sec, (int)(tRasEvent.tEvTime.tv_usec/1000),
				tRasEvent.nEvCode, tRasEvent.nEvParam1, tRasEvent.nEvParam2 );
		fflush( stdout );
	}

	return 0;
}
```

第７章　保守・障害解析関連

７．１　メモリダンプ機能

７．１．１　概要

この装置では、OSフリーズからの強制回復が発生した場合に、システムメモリの内容をメモリダンプファイルに記録（*1）、および障害初期調査・解析に利用する障害情報レポートを出力します。このとき、筐体前面ランプが下記状態表示されます。(*2)

- 筐体前面『状態表示デジタルLED』　　：「80」
- 筐体前面『状態識別LED』　　：「赤色／緑色／黄色　点灯」

（*1）メモリダンプの採取には Red Hat Enterprise Linux 6 OS機能の kdump を使用しています。kdump機能はメモリダンプ採取要因発生時、専用のカーネルを起動し、メモリダンプファイルを保存します。また、キー入力での採取も可能になります。

（*2）メモリダンプ採取処理に掛かる時間は、メモリ容量が4GBの場合は約3分程度かかる場合があります。

採取したメモリダンプファイル内容を解析することによって障害の原因を調査できます。メモリダンプの解析が必要な場合は、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX取扱説明書」 7.2メモリダンプ採取機能　を参照し、DVD-RAM等に保存した後、弊社担当営業までご連絡ください。

なお、メモリダンプの解析は有償となります。

７．１．２　メモリダンプの採取要因

この機能は、表７－１の要因でメモリダンプが採取されます。

表７－１　メモリダンプを採取する要因一覧

要因	説明
Linux カーネル障害	Linuxカーネル内で重要エラーが発生したとき、メモリダンプが採取されます。
キー入力	（[Alt]+[SysRq]+[c]キー）を入力した場合に採取されます。

なお、メモリダンプが発生した場合の対応については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「７．２　メモリダンプ採取機能」を参照ください。

７．１．３　メモリダンプファイル保存仕様

この機能が記録するメモリダンプファイル保存仕様を表７－２に示します。

このメモリダンプファイル保存結果はシステムログに記録されます。

表７－２　メモリダンプファイル保存仕様

No.	項目	内容
1	メモリダンプ保存用ディレクトリ	/var/crash/dump(.N)（*1）
2	メモリダンプファイル名	vmcore（*2）
3	障害情報レポートファイル名	report（*2）
4	メモリダンプファイル最大保存件数	2件

（*1）最新はdump、１世代前はdump.1となります。

（*2）保存に失敗した場合は、ファイル名の末尾に「-incomplete」が付加されます。

７．２　ログファイル収集機能

７．２．１　概要

ログファイル収集機能は、保守用のログファイルを収集する機能です。

raslogsaveコマンドにて収集することができます。

なお、raslogsaveコマンドは、情報取得時にシステムに負荷が掛かります。対象機器が本番系で負荷をかける事ができない場合のみ、オプションを指定し簡易ログを収集してください。通常は詳細ログを収集してください。（７．２．４　ログファイル収集コマンド（raslogsave）参照）

７．２．２　raslogsaveコマンド概要

raslogsaveコマンドは、システムに関する様々な情報をアーカイブファイルとして容易に一括収集できるツールです。以下のような情報を収集します。

詳細については、生成された圧縮ファイルを展開してご確認ください。

- システムのハードウェア情報
- 設定ファイル
- ログファイル
- コマンド実行時点のプロセス情報等

留意事項

a）情報取得時にはシステムに負荷が掛かります。
　システム稼動が行われていない時間帯などを考慮のうえ、適切なタイミングで実行してください。

b）情報取得にはメモリ容量が8GBの場合、Aモデル約1分程度、Dモデル約2分程度の時間が掛かります。
　ご使用環境により情報取得に時間が掛かる場合がありますのでシステム稼動の停止がある程度余裕をもって行える時間帯に実行してください。

c）実行後、中断は行わないでください。（Ctrl + c等により）中断した場合は、正常に処理が終了しない場合があります。

d）情報収集したファイルは、/tmp下に作成されます。約20MB程度のサイズになります。

e）できる限り多くのシステム情報を収集するため、取得情報の中にユーザ名、ホスト名、IP アドレス等が含まれます。
　※ /etc/passwd、/etc/shadow ファイル等は含まれません。
　※ 収集したファイル一覧は、下記コマンドで確認できます。
　　tar tf /tmp/raslog-YYYYMMDDhhmmss.tar （YYYYMMDDhhmmss = 年月日時分秒 ）

７．２．３　raslogsaveコマンド使用方法

raslogsaveコマンドによる、システム情報の採取手順を記載します。

raslogsaveコマンドを使用するには、root権限が必要です。root権限ユーザでログインしてから実行してください。

① root権限ユーザでログインしてください。

② raslogsaveコマンドを実行します。

```
# raslogsave
```

③ 下記表示があり情報収集が終了します。

/tmpディレクトリ下に raslog- YYYYMMDDhhmmss.tar が作成されます。

（YYYYMMDDhhmmss = 年月日時分秒 ）

```
raslogsave: INFO: raslogsave is finish.
                output file is /tmp/raslog- YYYYMMDDhhmmss.tar
```

留意事項

/tmpディレクトリ下にraslog- YYYYMMDDhhmmss.tar （YYYYMMDDhhmmss = 年月日時分秒 ）が作成されます。

障害発生時には必要に応じて上記ファイルをサポート窓口へ送付ください。

７．２．４　ログファイル収集コマンド（raslogsave）

＜名前＞

raslogsave　－　ログファイルを収集・圧縮して、格納

＜書式＞

raslogsave

＜実行権限＞

本コマンドは、root権限ユーザでのみ実行可能です。

＜機能＞

raslogsaveは、ログファイルを収集および圧縮するコマンドです。

詳細ログ収集を行い、tar形式でアーカイブ化します。

すでに起動中の場合、新たに起動はできません。

＜出力先＞

ログの収集時の一時ディレクトリは、/tmpディレクトリを使用します。

また、収集したログは、/tmpディレクトリに下記名称にて保存します。

raslog-YYYYMMDDhhmmss.tar　（YYYYMMDDhhmmss = 年月日時分秒 ）

それぞれの収集するログは、下記となります。

- システムのハードウェア情報
- 設定ファイル
- ログファイル
- コマンド実行時点のプロセス情報等

＜診断＞

処理が正常に終了した場合、戻り値に0を返します。異常終了した場合は戻り値に１を返します。

＜エラーメッセージ＞

No	エラーメッセージ	内容
1	raslogsave: ERROR: raslogsave is already running now.	多重起動した。
2	raslogsave: ERROR: Please　execute　root　user .	root権限のないユーザにて実行されている。
3	raslogsave: ERROR: /etc/rasutl/raslogsave.conf is not exist.	設定ファイルがない。
4	raslogsave: ERROR: OUTPUT_DIR(=/xx/xxx) is not exist.	ﾛｸﾞの出力先ﾃﾞｨﾚｸﾄﾘがない。
5	raslogsave: ERROR :xxx is not exist.	xxxが存在しない。
6	raslogsave: ERROR: /etc/rasutl/raslogsave.conf　is broken.	設定ファイルが壊れている。
7	raslogsave: ERROR: failed to sosreport command.	sosreportｺﾏﾝﾄﾞに失敗した。
8	raslogsave: ERROR: failed to move sosreport result to raslogsave dir.	sosreportをraslogsaveの作業ﾃﾞｨﾚｸﾄﾘに移動するのに失敗した。
9	raslogsave: ERROR: failed to compress log files.	tarｺﾏﾝﾄﾞによるアーカイブ化に失敗した。
10	raslogsave: ERROR: can't remove xxx	xxxﾃﾞｨﾚｸﾄﾘの削除に失敗した。

＜使用例＞

```
#raslogsave

raslogsave: INFO: sosreport is start. Please wait a few minutes
raslogsave: INFO: sosreport is end.
raslogsave: INFO: get hardware information is start. Please wait a few minutes
raslogsave: INFO: get hardware information is end.
raslogsave: INFO: get OS information is start. Please wait a few minutes.
.....

raslogsave: INFO: raslogsave is finish.
                  output file is /tmp/raslog-20130924110834.tar
```

sosreportの取得表示

留意事項

/tmpディレクトリ下に raslog-YYYYMMDDhhmmss.tar （YYYYMMDDhhmmss = 年月日時分秒 ）が作成されます。

障害発生時には必要に応じて上記ファイルをサポート窓口へ送付ください。

７．３　保守操作支援コマンド

保守操作支援コマンドの使用方法について説明します。これらのコマンドは装置でトラブルが発生したときや予防保全作業をするときに使用します。

表７－３に保守操作支援コマンドの一覧を示します。

表７－３　保守操作支援コマンド一覧

コマンド名	機能
raslogpt	RASイベントログの表示
rasconfig	RAS機能の設定情報表示／設定（*1）
rasstat	RAS機能が管理している装置状態の表示（*2）
raslogsave	ログファイルを収集・圧縮して格納（*3）
rasraidctl　NOTIFY	メディアエラー時の通知設定
rassimtool	RAS障害シミュレーションモードの開始／停止、 装置異常の模擬（*4）

（*1）「３．１　RAS機能設定コマンド（rasconfig）」を参照ください。

（*2）「４．１．１　RAS状態表示コマンド（rasstat）」を参照ください。

（*3）「７．２．４　ログファイル収集コマンド（raslogsave）」を参照ください。

（*4）「７．４．２　RAS障害シミュレーションコマンド（rassimtool）」を参照ください。

７．３．１　RASイベントログ表示コマンド（raslogpt）

＜名前＞

raslogpt － RASイベントログの表示

＜書式＞

raglogpt

＜実行権限＞

本コマンドは、全ユーザで実行可能です。

＜機能＞

raslogptは、RASイベントログを表示するためのコマンドです。

RASイベントログファイルに含まれる全てのログ情報を表示します。

＜オプション＞

なし

＜診断＞

処理が正常に終了した場合、exitステータス0を返します。異常終了した場合はexitステータス0以外を返します。

＜エラーメッセージ＞

なし

＜raslogptコマンド表示例＞

```
#
#
#raslogpt
[   1] 2013/04/25 14:02:19    [MODE]   Run mode was switched to SIMULATION.
[   2] 2013/04/25 14:03:00    [FAN]    CPUFAN slowed.
[   3] 2013/04/25 14:03:10    [MODE]   Run mode was switched to ONLINE.
[   4] 2013/04/25 14:06:36    [FAN]    SYSFAN returned normally.
[   5] 2013/04/25 14:07:21    [TEMP]   Temperature become too high.
#
#
#
#
#
```

７．３．２　メディアエラー時通知設定機能（rasraidctl　NOTIFY）(Dモデルのみ)

＜名前＞

rasraidctl　NOTIFY　－　メディアエラー時の通知設定を行います。

＜書式＞

rasraidctl　NOTIFY [ON | OFF]

＜実行権限＞

本コマンドは、root権限でのみ実行可能です。

＜機能＞

NOTIFYオプションは、メディアエラーが発生した場合に、MCALL接点をクローズ／オープンおよび状態表示デジタルLEDを表示／消去する設定を行うコマンドオプションです。
引数無しで実行した場合、現在の設定情報を表示します。
HF-BX/LXでは、[ON]設定の場合、メディアエラー発生時にMCALL接点はクローズ、状態表示デジタルLEDは表示となります。[OFF]設定の場合、MCALL接点はオープン、状態表示デジタルLEDは消去となります。
HF-BT/LXでは、[ON]設定の場合、メディアエラー発生時に状態表示デジタルLEDは表示となります。[OFF]設定の場合、状態表示デジタルLEDは消去となります。
本コマンドによる通知設定は、RASソフトウェア機能設定ファイルで管理されます。

＜オプション＞

[ON | OFF]オプションは下記のとおり

ON　：　メディアエラー時、MCALL 接点をクローズ、状態表示デジタル LED を表示します。（HF-BX/LX の場合）
　　メディアエラー時、状態表示デジタル LED を表示します。（HF-BT/LX の場合）

OFF　：　メディアエラー時、MCALL 接点をオープン、状態表示デジタル LED を表示します。（HF-BX/LX の場合）
　　メディアエラー時、状態表示デジタル LED を消灯します。（HF-BT/LX の場合）

なし　：　設定情報を表示します。

＜診断＞

処理が正常に終了した場合、exitステータス0を返します。異常終了した場合はexitステータス0以外を返します。

＜エラーメッセージ＞

表７－４　rasraidctl　NOTIFYコマンドのエラーメッセージ一覧

No	エラーメッセージ	内容
1	Usage: rasraidctl NOTIFY [ON|OFF]	オプションの指定に誤りがあります。
2	Access denied. errno=(xx)	root権限のないユーザにて実行されています。 ※xx＝エラー番号
3	ERROR: an error occurred while processing <xx>. errno=(yy),	予期せぬエラー(yy)が関数<xx>で発生しました。 ※xx=関数名yy=エラー番号
4	ERROR: an error occurred while processing <GetRasRunMode>. mode=(xx)	RAS機能動作モードがオフライン以外でありません。 ※xx＝RAS機能動作モード番号

＜**rasraidctl NOTIFYコマンド表示例**＞

```
#
#
# rasraidctl    NOTIFY ON
Setting was changed to ON.
#
#
#
#
# rasraidctl    NOTIFY OFF
Setting was changed to OFF.
#
#
#
# rasraidctl    NOTIFY
  MEDIA ERROR NOTIFY Setting : ON
#
```

・メディアエラー時、MCALL 接点をクローズ、状態表示デジタル LED を表示します。（HF-BX/LX の場合）
・メディアエラー時、状態表示デジタルLEDを表示します。（HF-BT/LXの場合）

・メディアエラー時、MCALL 接点をオープン、状態表示デジタル LED を消去します。（HF-BX/LX の場合）
・メディアエラー時、状態表示デジタルLEDを消去します。（HF-BT/LXの場合）

メディアエラー時、MCALL接点、状態表示デジタルLEDの設定を表示します。

７．４　RAS障害シミュレーション機能

本機能は、この装置のハードウェア状態をシミュレートします。

これにより、実際にハードウェア異常が発生していなくてもユーザアプリケーションのテストを実施することができます。

留意事項

RAS障害シミュレーション機能を使用中は、本装置のRAS機能は実際のRAS状態を監視していません。従って、本機能使用中では温度異常などを検出できない危険な状態にありますので、**業務では絶対に使用しないでください。**

本機能はユーザアプリケーションのテスト用途にのみ使用してください。また、RAS障害シミュレーション機能を使用後は、システムを再起動して実際にRAS障害が発生していないか確認してください。

ただし、RAS障害シミュレーション機能使用中も、実際にファン異常、温度異常が発生した場合は、メンテナンスコール接点（MCALL）がクローズします。

７．４．１　概要

RAS障害シミュレーション機能の機能概要は以下の通りです。

（１）rassimtoolコマンドによって擬似的に環境障害が発生した状態にできます。
　　実際にファン停止等は行いません。

（２）環境障害としては下記をサポートします。

- CPUファン障害
- ケースファン障害
- 温度障害
- RAID異常（Dモデルのみ）
- DRIVE異常

（３）環境障害を擬似的に発生させた後のメンテナンスコール接点（MCALL）　のクローズ、エラーメッセージ等の動作は、通常環境障害が発生した場合と同じです。

７．４．２　RAS障害シミュレーションコマンド（rassimtool）

＜名前＞

rassimtool　－　RAS障害シミュレーションモードの開始／停止、装置異常の模擬

＜書式＞

rassimtool　--help

rassimtool　[制御]

rassimtool　[模擬部位]　[模擬内容]

＜実行権限＞

本コマンドは、root権限でのみ実行可能です。

＜説明＞

rassimtoolは、RAS障害シミュレーションを行います。

装置異常の模擬を行うには、STARTオプションを使用してRAS障害シミュレーションモード状態にする必要があります。

RAS障害シミュレーションモードを終了するには、STOPオプションを使用します。

＜オプション＞

オプションは下記となります。

START	：	RAS 障害シミュレーションモード状態を開始します。 既に RAS 障害シミュレーションモード状態が開始されていた場合は 何もせずに終了します。
STOP	：	RAS 障害シミュレーションモード状態を終了します。 既に RAS 障害シミュレーションモード状態が終了されていた場合は 何もせずに終了します。
STATUS	：	RAS 障害シミュレーションモードの状態を表示します。

＜制限事項＞

本コマンドは root 権限ユーザ(ユーザ所属グループが root)のみが使用可能です。

ハードウェア異常が検出されている状態では、RAS 障害シミュレーション機能は開始できません。

未実装ハードウェア部位(ドライブ(A モデルのみ))を対象にした状態模擬設定は行えません。

[模擬部位]、[模擬内容]オプションは下記となります。

表７－５　rassimtool コマンドの[模擬部位]、[模擬内容]オプション一覧

No	模擬部位	模擬内容	説明	ﾓﾃﾞﾙ別ｻﾎﾟｰﾄ比較		備考
				Aモデル	Dモデル	
1	CPUFAN	NORMAL	CPU ファン回転状態正常	→	○	初期値
		ERROR	CPU ファン回転状態異常	→	○	
2	SYSFAN	NORMAL	ケースファン回転状態正常	→	○	初期値
		ERROR	ケースファン回転状態異常	→	○	
3	TEMP	NORMAL	筐体内温度状態正常	→	○	初期値
		ERROR	筐体内温度状態温度異常	→	○	
		DANGER	筐体内温度状態高温異常	→	○	
4	RAIDMEDIAERROR	NORMAL	RAID メディアエラーなし	×	○	初期値
		ERROR	RAID メディアエラーあり	×	○	
5	DRIVE1.STATUS	ONLINE	内蔵 RAID 構成ﾄﾞﾗｲﾌﾞ 1 状態正常	×	○	初期値
		OFFLINE	内蔵 RAID 構成ﾄﾞﾗｲﾌﾞ 1 状態ｵﾌﾗｲﾝ	×	○	
6	DRIVE1.SMART	NORMAL	内蔵ドライブ 1 SMART 検知なし	→	○	初期値
		ERROR	内蔵ドライブ 1 SMART 検知あり	→	○	
7	DRIVE1.OVERRUN	NORMAL	内蔵ドライブ 1 使用時間閾値超過なし	→	○	初期値
		ERROR	内蔵ドライブ 1 使用時間閾値超過あり	→	○	
8	DRIVE2.STATUS	ONLINE	内蔵 RAID 構成ﾄﾞﾗｲﾌﾞ 2 状態正常	×	○	初期値
		OFFLINE	内蔵 RAID 構成ﾄﾞﾗｲﾌﾞ 2 状態ｵﾌﾗｲﾝ	×	○	
9	DRIVE2.SMART	NORMAL	内蔵ドライブ 2SMART 検知なし	→	○	初期値
		ERROR	内蔵ドライブ 2SMART 検知あり	→	○	
10	DRIVE2.OVERRUN	NORMAL	内蔵ドライブ 2 使用時間閾値超過なし	→	○	初期値
		ERROR	内蔵ドライブ 2 使用時間閾値超過あり	→	○	

※RAIDアレイ状態の模擬について

RAIDアレイ状態は、ドライブ状態（DRIVEx.STATUS）の組み合わせにより、自動的に判別されます。組み合わせについては下表の内容となります。

表７－６　RAID アレイの状態 組合せ表

		DRIVE2 状態模擬内容	
		正常 (ONLINE)	オフライン (OFFLINE)
状態模擬内容 DRIVE1	正常 (ONLINE)	RAIDアレイ正常 (Optimal)	RAID アレイ縮退 (Degrade)
	オフライン (OFFLINE)	RAID アレイ縮退 (Degrade)	RAID アレイ故障 (Fail)

Optimal　: RAID状態が正常

Degrade　: RAID状態が縮退

Fail　　: RAID状態が異常

＜診断＞

処理が正常に終了した場合、exitステータス0を返す。異常終了した場合はexitステータス1以上を返します。

＜エラーメッセージ＞

表７－７　rassimtool コマンドのエラーメッセージ一覧

No	エラーメッセージ	説明
1	Usage: rassimtool START	オプションの指定に誤りがあります。 正しいオプションを指定してください。
2	Usage: (*1) rassimtool <START|STOP|STATUS> rassimtool <part> <status>	オプションの指定に誤りがあります。 正しいオプションを指定してください。 (*1)詳細内容は、--help オプション項を参照ください。
3	Access denied. errno=(*xxx*)	管理者権限がありません。 root ユーザまたは root グループに所属するユーザから実行してください。
4	Cannot switch to the simulation mode, because hardware error occured.	ハードウェア異常が発生しているため、RAS 障害シミュレーション機能を開始できません。
5	Currently, the ras mode is the online mode. switch to the simulation mode with "rassimtool START" command.	指定オプションは障害シミュレーションモード中のみ有効です。rassimtool START を実行し RAS 障害シミュレーション機能を開始してから再度実行してください。
6	Cannot simulate the state of [*XXX*], because [*XXX*] is not connected.	指定部位はハードウェアが実装されていないため、ハードウェア状態の模擬を行えません。(*XXX* =模擬部位)
7	ERROR: an error occurred while processing <*xxx*>.	予期せぬエラーが発生しました。再度コマンドを実行してください。それでもエラーメッセージが表示される場合は、本装置を再起動してください。

＜使用例＞

```
#
#
#rassimtool START          ←― RAS障害シミュレーションモード「開始」
RAS mode has been changed to [SIMULATION] mode.
#
#
#rassimtool STATUS         ←― rassimtool STATの表示例（Dモデル）
CPUFAN                    : ERROR
SYSFAN                    : NORMAL
TEMP                      : NORMAL
RAIDMEDIAERROR            : NORMAL   }←― Dモデル固有定義
DRIVE1.STATUS             : ONLINE   }
DRIVE1.SMART              : NORMAL
DRIVE1.OVERRUN            : ERROR
DRIVE2.STATUS             : ONLINE   ←― Dモデル固有定義
DRIVE2.SMART              : ERROR
DRIVE2.OVERRUN            : NORMAL
#
#
#
#rassimtool STATUS         ←― rassimtool STATの表示例（Aモデル）
CPUFAN                    : ERROR
SYSFAN                    : NORMAL
TEMP                      : NORMAL
DRIVE1.SMART              : NORMAL
DRIVE1.OVERRUN            : ERROR
DRIVE2.SMART              : ERROR
DRIVE2.OVERRUN            : NORMAL
#
#
#rassimtool STOP           ←― RAS障害シミュレーションモード「終了」
RAS mode has been changed to [ONLINE] mode.
#
```

このページは白紙です。

第８章　インストール方法

８．１　概要

本章では、RAS ユーティリティパッケージを本装置にインストールする方法について説明します。

留意事項

・本装置は工場出荷時に RAS ユーティリティパッケージをインストールしております。ご使用の際に改めてインストールする必要はありません。
・OS の再インストールを実施された場合は、下記手順にて RAS ユーティリティパッケージのインストールを行ってください。なお、OS の再インストール方法については、「HF-BX1000/LX HF-BX1200/LX 取扱説明書」または「HF-BT1000/LX 取扱説明書」の「９．２．２ 現在のシステムを廃棄して新規にインストールする手順」を参照してください。
・RAS ユーティリティパッケージのインストールは OS のインストール後、すぐに行ってください。また、インストール作業は途中で中断しないでください。

８．２　RASユーティリティパッケージのインストール

RASユーティリティパッケージをインストールするには、root権限が必要です。
root権限ユーザでログインしてから実行してください。

（１）root権限ユーザにてログインしてください。
（２）デスクトップ上で右クリックし、「端末の中に開く（E）」を選択しコンソール画面を開きます。
（３）本装置のDVDドライブに「RASUTL」のCD-ROMを挿入後、DVDドライブをマウントします。
※HF-BX1000/LX、HF-BX1200/LXの場合、DVDドライブは付属されていませんので、別途USB接続のDVDドライブをご用意いただきご使用ください。

コンソール画面上で下記の操作を行います。

```
# mkdir  –p  /media/rasutl
# mount  /dev/cdrom  /media/rasutl
# cd  /media/rasutl
```

（４）下記コマンドでインストールスクリプトを実行します。

```
# bash  setup-rasutl
```

インストールが完了し、下記のメッセージが表示されることを確認してください。
※インストールには約５分かかります。

```
+------------------------------------------------------------+
|  Please reboot a system.                                   |
|  Next time this system will start in runlevel 1.           |
+------------------------------------------------------------+
```

（５）下記コマンドを実行し本装置を再起動してください。

※「RASUTL」のCD-ROMは挿入したまま再起動ください。

shutdown　–r　now

再起動後、CUI 画面で起動します。

（６）OS起動後、CUI画面（ランレベル１）で立ち上がりますので以下の操作を実行し

Linuxライブラリなどのアップデートを実施します。

mount　/dev/cdrom　/media/rasutl

cd　/media/rasutl

bash　update-linux

インストールが完了し、下記のメッセージが表示されることを確認してください。

※インストールには約３分かかります。

```
+-----------------------------------------------------------+
|       Please reboot a system.                             |
+-----------------------------------------------------------+
```

（７）下記の操作でCD-ROMを取り出します。

cd　/

umount　/dev/cdrom

eject

（８）下記コマンドを実行し本装置を再起動してください。

shutdown　–r　now

再起動後、GUI 画面（ランレベル５）で起動することを確認してください。

以上で RAS ユーティリティパッケージのインストールは終了です。

８．３　RASソフトウェア機能で使用する設定ファイルとOSカスタマイズ用設定ファイル

RASソフトウェア機能で使用する各種設定ファイル、およびOSカスタマイズ用の設定ファイルは下記となります。

表８－１　RASソフトウェア機能の設定ファイル一覧

ファイル名	ファイル概略名称	格納ディレクトリ	ファイル概要
syscrashreport.conf	障害情報レポート 出力項目設定ファイル	/etc/rasutl	メモリダンプ機能の障害情報レポート出力項目情報を格納するファイル
snmpd.conf	snmpd デーモン設定 ファイル	/etc/snmp	RAS リモート機能で SNMP エージェントのアクセス制限を行う定義ファイル
snmpd	snmpd デーモン起動 オプション定義ファイル	/etc/sysconfig	SNMP 起動オプションの設定を行う定義ファイル